

## Beschreibung

Auswerteschaltung und Verfahren zum Feststellen und/oder zum Lokalisieren fehlerhafter Datenworte in einem Datenstrom  $T_n$

5

Integrierte Schaltungen, insbesondere schnelle digitale Schnittstellenschaltungen/Interfaces werden häufig schon während des Herstellungsverfahrens einem oder mehreren Produktionstests unterzogen, bei dem bzw. bei denen Testmuster/Testpattern an die integrierte Schaltung angelegt und der in Abhängigkeit dieser Testmuster von der integrierten Schaltung erzeugte Datenstrom untersucht werden.

: 10

Produktionstests, bei denen fehlerhafte integrierte Schaltungen zuverlässig entdeckt und rechtzeitig aussortiert werden können, dauern verhältnismäßig lange und erfordern einen hohen Aufwand.

15

Bei gängigen Produktionstests, bei denen aus Zeit- und Kostengründen die Datenströme komprimiert oder kompaktiert werden, ist es oft nicht möglich, fehlerhafte integrierte Schaltungen bereits während eines Produktionstests zu ermitteln, so dass fehlerhafte integrierte Schaltungen oft noch weitere Fertigungsstationen durchlaufen, bis sie als fehlerhaft identifiziert werden. Wenn fehlerhafte integrierte Schaltungen nicht während des oder der Produktionstests, sondern erst zu einem späteren Zeitpunkt im Herstellungsverfahren erkannt werden, entstehen bspw. bedingt durch die verringerte Produktionsausbeute hohe Kosten.

25

30

Wenn bei mit komprimierten oder kompaktierten Datenströmen arbeitenden Produktionstests ein fehlerhafte Schaltung erkannt worden ist, so kann noch keine Aussage darüber getroffen wer-

den, welche Stelle oder welcher Bereich diesen Fehler verursacht hat. Dies muss durch Aussortieren der fehlerhaften Schaltung sowie durch einen separaten Testlauf festgestellt werden.

5

Es ist Aufgabe der vorliegenden Erfindung, eine Vorrichtung sowie ein Verfahren bereitzustellen, mit dem bzw. mit der eine produktionsbegleitende Überprüfung von integrierten Schaltungen ermöglicht wird, bei der vorhandene Fehler in den getesteten Schaltungen zuverlässig erkannt und genau lokalisiert werden können.

Diese Aufgabe wird mit dem Gegenstand der unabhängigen Patentansprüche gelöst. Vorteilhafte Weiterbildungen der Erfindung ergeben sich aus den jeweiligen Unteransprüchen.

Die erfindungsgemäße Auswerteschaltung ist zum Feststellen und/oder zum Lokalisieren von fehlerhaften Datenworten in einem Datenstrom  $T_n$  vorgesehen. Sie weist eine erste lineare Automaten-schaltung sowie eine parallel geschaltete zweite lineare Automaten-schaltung mit jeweils einer Menge von Zuständen  $z(t)$  auf. Bei diesen linearen Automaten-schaltungen handelt es sich insbesondere um linear rückgekoppelte Schieberegistern mit einer Multiple-Input-Linear-Feedback-Shift-Register-Architektur/MILFSR- Architektur.

Beide lineare Automaten-schaltungen sind mit einer gemeinsamen Eingangsleitung zur Aufnahme eines Datenstroms  $T_n$  aus  $n$  aufeinanderfolgenden, jeweils  $k$  Bit breiten Datenwörtern  $y(1)$ , ...,  $y(n)$  ausgestattet. An diese Eingangsleitung können beliebige Testdaten angelegt werden, wobei die Gutsignatur der idealen fehlerfreien Testdaten bekannt sein muss. Die erste lineare Automaten-schaltung ist durch die Gleichung

$$z(t+1) = Az(t) \oplus y(t)$$

und die zweite lineare Automatenschaltung durch die Gleichung.

5

$$z(t+1) = Bz(t) \oplus y(t)$$

beschrieben. Dabei stellen A und B die Zustandsmatrizen der linearen Automatenschaltungen dar.

10

Die beiden linearen Automatenschaltungen können eine erste Signatur S1 bzw. eine zweite Signatur S2 berechnen, entweder direkt aus den Datenwörtern  $y(1), \dots, y(n)$  des Datenstroms  $T_n$  oder ggf. aus bereits kodierten Datenwörtern  $u^1(1), \dots, u^1(n)$  bzw.  $u^2(1), \dots, u^2(n)$ .

15

Unter Signatur wird in diesem Dokument eine Kompaktierung einer Menge von Datenwörtern verstanden. Aus diesen Signaturen ist es möglich, auf die fehlerhaften Datenwörter zurückzurechnen.

20

Diese berechneten Signaturen S1 und S2 werden durch ein nach der ersten linearen Automatenschaltung angeordnetes erstes Verknüpfungsgatter sowie durch ein nach der zweiten linearen Automatenschaltung angeordnetes zweites Verknüpfungsgatter jeweils mit einer fehlerfreien Gutsignatur verglichen. An den Ausgängen der Verknüpfungsgatter kann ein Vergleichswert mit den Gutsignaturen abgegriffen werden, mittels dessen ein Rückschluss darauf möglich ist, ob der betrachtete Datenstrom  $T_n$  kein, ein oder mehrere fehlerhafte Datenwörter  $y'(i)$  aufweist.

25

30

Gemäß einem Grundgedanken der Erfindung kann aus den Signaturen eines Datenstroms  $T_n$  somit direkt auf die Anzahl der feh-

lerhaften Datenworte  $y'(i)$  im Datenstrom  $T_n$  geschlossen werden.

Falls im Datenstrom  $T_n$  genau ein fehlerhaftes Datenwort  $y'(i)$  in der  $i$ -ten Position des Datenstromes vorhanden ist, das sich von dem fehlerfreien Datenwort  $y(i)$  um das nachfolgend auch einfach als Fehler bezeichnete Fehlerwort  $e(i)$ ,  $y'(i) = y(i) \oplus e(i)$ , unterscheidet, wobei mit  $\oplus$  die komponentenweise Addition modulo 2 bezeichnet ist, ist es erfindungsgemäß möglich, die Position  $i$  des fehlerhaften Datenwortes  $y'(i)$  im Datenstrom und den Fehler  $e(i)$  direkt aus den Differenzen der Signaturen  $S_1$  und  $S_2$  von den entsprechenden fehlerfreien Signaturen zu bestimmen.

Die bei einem Produktionstest ständig an eine Ausgabeeinheit zu übertragenden und zu untersuchenden Testmuster reduzieren sich daher erfindungsgemäß um mehrere Größenordnungen. Es ist für die Fehlerlokalisierung nicht mehr nötig, einen Vergleich aller Datenworte mit den jeweils bekannten fehlerfreien Datenworten, also insgesamt 2 mal  $n$  Datenworten durchzuführen.

Es ist ebenfalls möglich, eine produktionsbegleitende Fehlerstatistik zu führen und auszuwerten, um festzustellen, ob fehlerhafte Bauteile durch einen derart reduzierten Datenstrom zu detektieren sind.

In einer ersten Ausführungsform der Auswerteschaltung, sind die Verknüpfungsgatter als exklusive Oder-Gatter ausgebildet, deren erster Eingang jeweils mit dem Ausgang der zugehörigen linearen Automaten-schaltung verbunden ist und an deren zweiten Eingang eine Gutsignatur anzulegen ist.

- Gemäß der Erfindung wird der Datenstrom  $T_n$  aus  $n$  Datenworten in zwei unterschiedlichen, linear rückgekoppelten Schieberegistern mit  $k$  parallelen Eingängen zu zwei Signaturen  $S1$  und  $S2$  kompaktiert. Aus den beiden bekannten Gutsignaturen  $GS1$  und  $GS2$  der beiden Schieberegister für den fehlerfreien Datenstrom  $T_n$  ergeben sich durch eine exklusive Oder-Verknüpfung mit den tatsächlich bestimmten Signaturen  $S1$  und  $S2$  die Signaturdifferenzen  $\Delta S1 = GS1 \oplus S1$  und  $\Delta S2 = GS2 \oplus S2$ . Aus diesen Signaturdifferenzen  $\Delta S1$  und  $\Delta S2$  sind für den Fall, dass nur ein Datenwort im Datenstrom fehlerhaft ist, die Position  $i$  des fehlerhaften Datenwortes  $y'(i)$  im Datenstrom  $T_n$  und das Fehlerwort  $e(i)$ , das die Abweichung des fehlerhaften Datenwortes  $y'(i)$  vom korrekten Datenwort  $y(i)$ ,  $y'(i) = y(i) \oplus e(i)$  beschreibt, bestimmt.
- 15 Liegen Fehler in zwei oder mehreren, beliebig vielen Datenworten vor, so ergibt sich aus den Signaturdifferenzen  $\Delta S1$  und  $\Delta S2$  als Ergebnis der erfindungsgemäßen Auswerteschaltung, dass der Datenstrom  $T_n$  fehlerhaft ist.
- 20 In einer zweiten Ausführungsform der Erfindung verfügt die Auswerteschaltung über einen vor der ersten linearen Automata-schaltung angeordneten ersten Kodierer. Dieser kodiert das Datenwort  $y(i)$  mit der Datenwortlänge von  $k$  Bit für  $i=1, \dots, n$  in ein kodiertes Datenwort  $u^1(i)$ ,  $u^1(i) = \text{Cod1}(y(i))$  der Wortbreite von  $K1$  Bit.  $\text{Cod1}$  ist dabei die Kodierungsfunktion des ersten Kodierers.
- 25 Dabei kann die Kodierfunktion  $\text{Cod1}$  des ersten Kodierers so beschaffen sein,
- 30 - dass für  $y'(i) = y(i) \oplus e(i)$  eine Funktion  $f_1$  mit  $f_1(0) = \bar{0}$  existiert,

- dass  $\text{Cod1}(y'(i)) = \text{Cod1}(y(i) \oplus e(i)) = \text{Cod1}(y(i) \oplus f_1(e(i)))$  bzw.  $\text{Cod1}(y'(i)) = u^1(i) \oplus f_1(e(i))$  gilt
- und dass es eine Funktion  $f_1^{-1}$ , mit  $f_1^{-1}(f_1(e))=e$  für alle möglichen  $k$ -stelligen Binärwörter  $e$  gibt, wobei  $e$  ein Fehlerwort ist, um das ein fehlerhaftes Datenwort im Datenstrom  $T_n$  von einem korrekten Datenwort abweichen kann.

In einer weiteren Ausführungsform der Erfindung verfügt die Auswerteschaltung über einen vor der zweiten linearen Automaten-  
 10 tenschaltung angeordneten zweiten Kodierer. Dieser kodiert das Datenwort  $y(i)$  mit der Datenwortlänge von  $k$  Bit für  $i=1, \dots, n$  in ein kodiertes Datenwort  $u^2(i)$ ,  $u^2(i)=\text{Cod2}(y(i))$  der Wortbreite von  $K_2$  Bit. Dabei ist  $\text{Cod2}$  die Kodierungsfunktion des zweiten Kodierers.

15

Die Kodierfunktion  $\text{Cod2}$  des zweiten Kodierers kann dabei so beschaffen sein,

- dass für  $y'(i)$  gilt  $y'(i) = y(i) \oplus e(i) = \text{Cod2}(y(i)) \oplus f_2(e(i))$  bzw.  $\text{Cod2}(y'(i)) = u^2(i) \oplus f_2(e(i))$
- 20 - und dass eine Funktion  $f_2^{-1}$  mit  $f_2^{-1}(f_2(e))=e$  existiert.

Die Kodierer können auch als durchgezogene Leitungen realisiert sein und eine identische Abbildung bewirken:

25  $\text{Cod1}(y(i))=\text{Cod2}(y(i))=y(i)$  für  $i=1, \dots, n$

Diese Ausgestaltung entspricht der bereits beschriebenen Auswerteschaltung ohne Kodierer.

30 Die Auswertung wird vereinfacht, wenn die Wortbreite  $K_1$  der von dem ersten Kodierer kodierten Datenwörter  $u^1(i)$  gleich der Wortbreite  $K_2$  der von dem zweiten Kodierer kodierten Datenwör-



ter  $u^2(i)$  ist, wenn der erste Kodierer hinsichtlich seines Aufbaus und seiner Funktion mit dem zweiten Kodierer übereinstimmt und wenn die Wortbreite  $K1$  der von dem ersten Kodierer kodierten Datenwörter  $u^1(i)$  und die Wortbreite  $K2$  der von dem zweiten Kodierer kodierten Datenwörter  $u^2(i)$  jeweils gleich der Wortbreite  $k$  der Datenwörter  $y(1), \dots, y(n)$  des Datenstroms  $T_n$  ist.

In einer vorteilhaften Weiterbildung der Erfindung können die Kodierungsfunktionen  $Cod1$  und  $Cod2$  des ersten Kodierers und des zweiten Kodierers wie folgt ausgebildet sein:

$$\begin{aligned} &Cod1(y_1(i), y_2(i), \dots, y_k(i)) \\ &= P1(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0) \end{aligned}$$

$$\begin{aligned} &Cod2(y_1(i), y_2(i), \dots, y_k(i)) \\ &= P2(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0) \end{aligned}$$

für  $i, 1, \dots, n$

20

Dabei ist die Anzahl der am Ende von  $P1(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  befindlichen Nullen gleich  $(K1-k)$ , und die Anzahl der am Ende von  $P2(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  ist gleich  $(K2-k)$ .  $P1$  stellt eine beliebige Permutation der  $K1$  Komponenten von  $(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$ , und  $P2$  eine beliebige Permutation der  $K2$  Komponenten von  $(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  dar.

In die  $K1=K2=k+1$  Bit breiten Zustandsvektoren der linearen Automatenschaltungen werden in diesem Fall nur  $k$  Bit von  $y_1(i), \dots, y_k(i)$  über XOR-Elemente eingegeben. Für die Eingabe der konstanten Werte 0 sind keine XOR-Elemente erforderlich. Da-

30

durch wird eine einfache Anpassung der Wortbreite der Datenwörter an die Wortbreite der Zustände gewährleistet.

Praktisch werden dabei in das k-Bit Datenwort  $y_1(i), \dots, y_k(i)=y(i)$  an  $(K1-k)$  Stellen bzw. an  $(K2-k)$  Stellen Nullen eingefügt. Die Komponenten können dabei durch die Permutationen noch vertauscht werden. Dadurch wird die Dimension der linearen Automaten-schaltungen gleich  $K1 > k, K2 > k$ . Die Wahrscheinlichkeit eines falsch erkannten Fehlers wird damit geringer.

In einer alternativen Ausprägung der Auswerteschaltung sind die Kodierungsfunktionen Cod1 und Cod2 des ersten Kodierers und des zweiten Kodierers wie folgt ausgebildet:

$$\begin{aligned} \text{Cod1}(y_1(i), y_2(i), \dots, y_k(i)) \\ = P1(y_1(i), y_2(i), \dots, y_k(i), b_1^1, \dots, b_{K1-k}^1) \end{aligned}$$

$$\begin{aligned} \text{Cod2}(y_1(i), y_2(i), \dots, y_k(i)) \\ = P2(y_1(i), y_2(i), \dots, y_k(i), b_1^2, \dots, b_{K2-k}^2) \end{aligned}$$

mit  $b_1^1, \dots, b_{K1-k}^1, b_1^2, \dots, b_{K2-k}^2 \in \{0,1\}$ . P1 und P2 stellen dabei beliebige Permutationen dar. Bei der praktischen Umsetzung dieser Ausprägung der Erfindung werden in das Datenwort  $y_1(i), \dots, y_k(i)=y(i)$  an  $K1-k$  ( $K2-k$ ) Stellen Nullen und Einsen eingefügt.

Besonders vorteilhaft ist es, wenn die Kodierungsfunktion Cod1 des ersten Kodierers und/oder die Kodierungsfunktion Cod2 des zweiten Kodierers so ausgebildet sind, dass sie einen linearen Blockcode,  $f_1=\text{Cod1}$  bzw.  $f_2=\text{Cod2}$ , realisieren. Dem Fachmann bekannte lineare Blockcodes können bspw. als Hamming-Kodes, als Paritätsbit-Kodes oder als Gruppen-Paritätsbit-Kodes ausgebildet sein.



Das Feststellen und das Lokalisieren fehlerhaften Datenworte in einem Datenstrom  $T_n$  kann dadurch vereinfacht werden, indem die linearen Automatenschaltungen so gewählt werden, dass ihre Zustandsmatrizen A und B wie folgt miteinander in Beziehung stehen:

$$B = A^n, \text{ mit } n \neq 1,$$

oder wenn die Zustandsmatrix B der zweiten linearen Automatenschaltung gleich der invertierten Zustandsmatrix  $A^{-1}$  der ersten linearen Automatenschaltung ist

$$B = A^{-1}$$

15

In einer weiteren Ausführungsform der Auswerteschaltung liegen die erste lineare Automatenschaltung als linear rückgekoppeltes Schieberegister und die zweite lineare Automatenschaltung als inverses linear rückgekoppeltes Schieberegister vor. Dabei weisen beide lineare Automatenschaltungen eine parallele Eingabe auf.

20

Die linearen Automatenschaltungen können auch als linear rückgekoppelte,  $K_1$ -dimensionale bzw.  $K_2$ -dimensionale Multi-Input-Schieberegister, insbesondere maximale Längen vorliegen.

25

Die Erfindung betrifft auch ein Verfahren zum Feststellen fehlerhafter Datenworte  $y'(i)$  in einem Datenstrom  $T_n$  aus  $n$  Datenwörtern  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  der Datenwortbreite  $k$  und/oder zum Lokalisieren eines fehlerhaften Datenworts  $y'(i)$  und einer fehlerhaften Position  $i$  eines fehlerhaften Datenworts  $y'(i) = y(i) \oplus e(i)$ .

30

Der Datenstrom  $T_n$  weicht dabei in der  $i$ -ten Position von dem korrekten Datenstrom  $y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)$  um  $e(i)$  ab.

5

Dabei werden zunächst die Datenwörter des Datenstromes  $T_n$  in einen ersten Kodierer mit der Kodierfunktion  $Cod1$  und der Dekodierungsfunktion  $Decod1$  eingegeben. Der erste Kodierer kodiert die Datenwörter  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  in die kodierten Datenwörter  $u^1(1), \dots, u^1(i-1), u^{1'}(i), u^1(i), \dots, u^1(n)$  der Wortbreite  $K1$  mit  $K1 \geq k$ .

10

Für  $y'(i) = y(i) \oplus e(i)$  existiert eine Funktion  $f_1$  derart, dass  $f_1(0)=0$  und  $Cod1(y'(i)) = Cod1(y(i) \oplus e(i)) = Cod1(y(i) \oplus f_1(e(i)))$  gilt und dass es eine Funktion  $f_1^{-1}(f(e))=e$  für alle prinzipiell möglichen  $k$ -stelligen Fehler mit  $f_1^{-1}(f_1(e))$  gibt.

15

Dann werden die derart kodierten Datenwörter in die Eingänge einer ersten linearen Automatenschaltung mit  $K1$ -dimensionalem Zustandsvektor  $z^1$  eingegeben.

20

Die erste lineare Automatenschaltung ist durch die Automaten-gleichung

25

$$z^1(t+1) = A \cdot z^1(t) + u^1(t) \quad (I)$$

beschrieben. Ihre Matrix  $A$  ist als  $K1 \times K1$  Matrix mit binärem Koeffizienten ausgebildet, so dass eine inverse Matrix  $A^{-1}$  existiert und die Additionen und Multiplikationen in (1) modulo 2 erfolgen.

: 30

Die erste lineare Automatenschaltung geht bei Eingabe von  $y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)$  in den Kodierer und damit bei Eingabe von  $u^1(1), \dots, u^1(i-1), u^1(i), u^1(i+1), \dots, u^1(n)$  in die erste lineare Automatenschaltung in den Zustand

5  $z^1(n+1) = S(L1, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  über.

Bei Eingabe der in der  $i$ -ten Position fehlerhaften Eingabe  $y(1), \dots, y(i-1), y'(i), \dots, y(n)$  in den Kodierer und damit

10 bei Eingabe von  $u^1(1), \dots, u^1(i-1), u^{1'}(i), \dots, u^1(n)$  in die erste lineare Automatenschaltung geht diese in den Zustand  $z^{1'}(n+1) = S(L1, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$  über.

15 Dabei sind die Signaturen der korrekten und der fehlerhaften Datenfolge jeweils mit  $S(L1, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  und mit  $S(L1, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$  bezeichnet.

20 Diese Signatur des Datenstroms  $T_n$  kann mit einer bekannten Gutsignatur verglichen werden. Ergibt dieser Vergleich keine auf wenigstens ein fehlerhaftes Datenwort  $y'(i)$  hindeutende Abweichung, so ist der Datenstrom  $T_n$  mit hoher Wahrscheinlichkeit fehlerfrei. In diesem Fall wird das erfindungsgemäße Ver-

25 fahren mit einem neuen Datenstrom  $T_n$  wieder von vorne begonnen.

Wird hingegen eine auf wenigstens ein fehlerhaftes Datenwort  $y'(i)$  hindeutende Abweichung festgestellt, so fährt das Ver-

30 fahren mit dem nächsten Verfahrensschritt fort, bei dem die Datenwörter  $y(1), \dots, y(i-1), y'(i), \dots, y(n)$  des Datenstromes  $T_n$  in einem zweiten Kodierer mit der Kodierungsfunktion  $Cod2$  und der Dekodierungsfunktion  $Decod2$  eingegeben werden.

Die Eingabe des Datenstromes  $T_n$  in den zweiten Kodierer erfolgt in der Praxis zumeist zeitgleich mit der Eingabe in den ersten Kodierer. Wenn der gleiche Datenstrom  $T_n$  auch zweimal  
 5 zeitversetzt zur Verfügung gestellt werden kann, kann die Eingabe in die Kodierer auch nacheinander erfolgen. Bei dieser Ausführungsform ist von Vorteil, dass nur ein Kodierer und nur eine lineare Automatenschaltung zur Verfügung stehen muss, die zweimal hintereinander für den gleichen Datenstrom  $T_n$  zum Einsatz  
 10 kommen.

Der zweite Kodierer und die zweite Automatenschaltung können dabei durch eine einfache Modifikation aus dem ersten Kodierer und aus der ersten Automatenschaltung erhalten werden.

15

Der zweite Kodierer kodiert die Datenwörter  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  in die kodierten Datenwörter  $u^2(1), \dots, u^2(i-1), u^{2'}(i), u^2(i+1)$  der Datenwortbreite  $K_2, K_2 \geq k$ .

20 Für  $y'(i)$  gilt dabei

$$\text{Cod}_2(y'(i)) = \text{Cod}_2(y(i) \oplus e(i)) = \text{Cod}_2(y(i)) \oplus f_2(e(i))$$

Es existiert eine Funktion  $f_2^{-1}$  mit  $f_2^{-1}(f_2(e)) = e$ .

25

Die derart kodierten Datenwörter werden in die Eingänge einer zweiten linearen Automatenschaltung mit  $K_2$ -dimensionalem Zustandsvektor  $z^2$  eingegeben. Dieser ist durch die folgende Automatengleichung beschrieben:

30

$$z^2(t+1) = B \cdot z^2(t) \oplus u^2(t) \tag{VII}$$

Die Matrix  $B$  der zweiten linearen Automatenschaltung mit  $B \neq A$  ist eine  $K_2 \times K_2$ -Matrix mit binärem Koeffizienten, für die eine inverse Matrix  $B^{-1}$  existiert. Die Additionen und Multiplikationen in (VII) erfolgen modulo 2.

5

Die zweite lineare Automatenschaltung geht bei der Eingabe der korrekten Datenfolge  $y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)$  in den zweiten Kodierer und bei Eingabe der kodierten Folge  $u^1(1), \dots, u^1(i-1), u^1(i), u^1(i+1), \dots, u^1(n)$  in den Zustand  $z^2(n+1) = S(L_2, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  über.

Bei Eingabe der fehlerhaften Datenfolge  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  in den zweiten Kodierer und bei Eingabe der kodierten Folge  $u^2(1), \dots, u^2(i-1), u^2(i), u^2(i+1), \dots, u^2(n)$  in die zweite lineare Automatenschaltung geht diese in den Zustand  $z^{2'}(n+1) = S(L_2, y(1), \dots, y(i-1), y(i), y'(i), y(i+1), \dots, y(n))$  über.

20 Dabei sind die Signaturen der korrekten und der fehlerhaften Datenfolge jeweils mit  $S(L_2, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  und mit  $S(L_2, y(1), \dots, y(i-1), y'(i), \dots, y(n))$  bezeichnet.

25 Nun können die Signaturdifferenzen  $\Delta S_1$  und  $\Delta S_2$  bestimmt werden, und zwar aus den bestimmten Signaturen  $S_1$  und  $S_2$  und aus den vorgegebenen Gutsignaturen. Diese Berechnung der Signaturdifferenzen  $\Delta S_1$  und  $\Delta S_2$  erfolgt bspw. durch komponentenweise exklusive Oder-Verknüpfungen der Signaturen  $S_1$  und  $S_2$  mit den  
30 vorgegebenen Gutsignaturen.

Die Signaturdifferenzen  $\Delta S_1$  und  $\Delta S_2$  sind wie folgt definiert.

$$\Delta S1 = S(L1, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)) \oplus \\ S(L1, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$$

und

5

$$\Delta S2 = S(L2, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)) \oplus \\ S(L2, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$$

Dabei werden die Signaturen komponentenweise XOR verknüpft.

10

Die folgenden Verfahrensschritte können von einer externen Berechnungseinheit, bspw. einem Computersystem unter Verwendung der bis zu diesem Zeitpunkt des Verfahrens bestimmten Werte vorgenommen werden.

15

Nun kann, falls genau ein fehlerhaftes Datenwort vorliegt, auf dessen Position  $i$  zurückgerechnet werden, und zwar indem eine eindeutige Lösung für die Position  $i$  des fehlerhaften Datenworts durch Lösen der Gleichung

20

$$f_1^{-1}(A^{i-n} \Delta S1) = f_2^{-1}(B^{i-n} \Delta S2) \quad (\text{XVI})$$

bestimmt wird. Der Wert der Position  $i$  kann aus der Gleichung (XVI) eindeutig bestimmt werden.

25

Falls sich keine eindeutige Lösung für  $i$  ergibt, wird durch ein Ausgabemedium eine Mitteilung ausgegeben, dass in dem betrachteten Datenstroms  $T_n$  zwei oder mehr Fehler vorliegen.

30

Wenn der Wert der Position  $i$  berechnet worden ist, so kann auch eine eindeutige Lösung für alle Bits des fehlerhaften Datenworts  $e(i)$  im Datenstroms  $T_n$  bestimmt werden, und zwar durch Lösen der Gleichung



$$e(i) = f_1^{-1}(A^{i-n} \cdot \Delta S1) \quad (\text{XIV})$$

Der Wert für  $e(i)$  ist aus der Gleichung (XIV) für den aus  
5 (XVI) bestimmten Wert für  $i \in \{1, \dots, n\}$  festgelegt.

Die Position  $i$  des fehlerhaften Datenworts  $y'(i)$  sowie der Fehler  $e$  im Datenstrom  $T_n$  können durch ein Ausgabemedium, bspw. durch einen Bildschirm oder durch einen Drucker zur Ver-  
10 fügung gestellt werden.

Die Verfahrensschritte des vorstehend beschriebenen Verfahrens können mit einer bereits beschriebenen, erfindungsgemäßen Auswerteschaltung durchgeführt werden, deren Ausgänge, an denen  
15 die Signaturdifferenzen  $\Delta S1$  und  $\Delta S2$  bereitgestellt werden, ggf. mit einer externen Berechnungseinheit, bspw. einem Computersystem verbunden ist, das die Rückberechnungsschritte durchführt.

20 Die Erfindung betrifft auch eine zu testenden integrierte Schaltung, auf der eine erfindungsgemäße Auswerteschaltung in einer der vorstehend beschriebenen Ausführungsformen, insbesondere zusätzlich zur normalen Schaltung quasi als add-on enthalten ist. Dabei ist die erfindungsgemäße Auswerteschal-  
25 tung auf dem integrierten Schaltkreis oder auf dem Halbleiterbauteil monolithisch integriert.

Die Erfindung betrifft auch eine Nadelkarte zum Testen von integrierten Schaltungen, bei der eine erfindungsgemäße Auswerteschaltung in einer der vorstehend beschriebenen Ausführungs-  
30 formen integriert ist.

Die Erfindung betrifft weiterhin ein testerspezifisches load board mit Testfassungen zum Einstecken von integrierten Schaltungen oder zur Aufnahme einer solchen Nadelkarte oder zum Anschluss eines handlers, wobei auf dem load board wenigstens  
5 eine erfindungsgemäße Auswerteschaltung in einer der vorstehend beschriebenen Ausführungsformen integriert ist. Ein solches load board kann auch als Adapterboard bezeichnet werden.

Die erfindungsgemäße Auswerteschaltung kann auch direkt auf  
10 einem Tester/ Messgerät/Testsystem/Testautomat zum Testen von integrierten Schaltkreisen integriert werden. Solch ein Tester verfügt über mehrere Instrumente zum Erzeugen von Signalen oder Datenströmen und über mehrere Mess-Sensoren, insbesondere für Ströme und Spannungen und weist ein Loadboard auf, das zur  
15 Aufnahme wenigstens einer Nadelkarte zum Testen von integrierten Schaltkreisen und/oder zum Anschluss eines Handlers an einen Tester von integrierten Schaltkreisen vorgesehen ist und/oder das mit wenigstens einem Testsockel zum Testen von integrierten Schaltkreisen ausgestattet ist.

20

Gemäß einem weiteren Grundgedanken der Erfindung kann die erfindungsgemäße Auswerteschaltung in allen vorstehend beschriebenen Ausführungsformen einfach und sehr platzsparend auf allen möglichen Schaltungen oder Geräten in jeder Abstraktionsebene bzw. auf jeder Messgeräteebene vorgesehen werden. Beeinträchtigungen der Funktionsweise ergeben sich dabei nicht. Die konkrete Ausgestaltung der vorstehend beschriebenen Gegenstände mit einer solchen Auswerteschaltung ergibt sich für den Fachmann vollständig und eindeutig aus den in dieser Patentschrift enthaltenen Informationen sowie aus seinem Fachwissen.  
30 Dabei ist lediglich zu beachten, dass die erfindungsgemäße Auswerteschaltung jeweils zusätzlich zu den auf den vorstehend

genannten Gegenständen enthaltenen Schaltungen aufzubringen ist.

Die Erfindung wird auch in einem Computerprogramm zum Ausführen des Verfahrens zum Feststellen und/oder zum Lokalisieren fehlerhafter Datenworte in einem Datenstrom  $T_n$  verwirklicht. Das Computerprogramm enthält dabei Programmanweisungen, die ein Computersystem veranlassen, ein solches Testverfahren in einer vorstehend beschriebenen Ausführungsform auszuführen.

10

Dabei werden insbesondere die Verfahrensschritte des Erzeugens und des Eingebens von Datenwörtern  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  eines Datenstroms  $T_n$  in die Auswerteschaltung sowie die Rückberechnungsschritte auf die Position des fehlerhaften Datenworts  $y'(i)$  und des Fehlers  $e$  aus den Signaturdifferenzen  $\Delta S1$  und  $\Delta S2$  mit einem Computersystem gesteuert oder auf einem Computersystem selbst durchgeführt. Das Computerprogramm gibt die Ergebnisse als digitale Datenfolgen oder in einer daraus generierten Darstellungsform auf einer Ausgabeeinheit aus, insbesondere auf einem Bildschirm oder auf einem Drucker, oder speichert diese Ergebnisdaten in einem Speicherbereich. Durch das erfindungsgemäße Computerprogramm können fehlerhafte Datenworte schnell, effektiv und zuverlässig bestimmt werden, wobei sich eine deutliche Beschleunigung der Testlaufzeit ergibt.

25

Die Erfindung betrifft außerdem ein Computerprogramm, das auf einem Speichermedium, insbesondere in einem Computerspeicher oder in einem Direktzugriffsspeicher enthalten ist oder das auf einem elektrischen Trägersignal übertragen wird. Die Erfindung betrifft auch ein Trägermedium, insbesondere einen Datenträger, wie bspw. eine Diskette, ein Zip-Laufwerk, einen Streamer, eine CD oder eine DVD, auf denen ein vorstehend be-

30

schriebenes Computerprogramm abgelegt ist. Ferner betrifft die Erfindung ein Computersystem, auf dem ein solches Computerprogramm gespeichert ist. Schließlich betrifft die Erfindung auch ein Download-Verfahren, bei dem ein solches Computerprogramm  
5 aus einem elektronischen Datennetz, wie bspw. aus dem Internet, auf einen an das Datennetz angeschlossenen Computer heruntergeladen wird.

Die Erfindung ist in den Zeichnungen anhand eines Ausführungs-  
10 beispiels näher veranschaulicht.

- Figur 1 zeigt eine erste Messdatenflussdarstellung mit  $n$  fehlerfreien Datenworten  $y(1), y(2), \dots, y(n)$ ,  
Figur 2 zeigt eine zweite Messdatenflussdarstellung mit einem fehlerhaften Messdatenwort  $y'(i)$ ,  
15 Figur 3 zeigt eine schematische Darstellung einer ersten linearen Automaten-schaltung,  
Figur 4 zeigt eine schematische Darstellung einer zweiten linearen Automaten-schaltung,  
20 Figur 5 zeigt ein Ablaufschema zum Bestimmen von Signaturunterschieden  $\Delta S1$  und  $\Delta S2$  aus einem Messdatenstrom  $T_n$ ,  
Figur 6 zeigt ein Ablaufdiagramm zur Veranschaulichung des erfindungsgemäßen Verfahrens zum Feststellen eines Fehlers  $e(i)$  in einem Datenwort  $y(i)$ .

25

Figur 1 zeigt eine erste Messdatenflussdarstellung 1.

Die erste Messdatenflussdarstellung 1 umfasst einen in Figur 1 rechteckig dargestellten pseudo-zufälligen Messdatenstrom  $T_n$ ,  
30 der sich in  $n$  aufeinanderfolgende Datenworte  $y(1), y(2), \dots, y(n)$  mit jeweils einer gleichen Datenwortlänge von  $k$  Bit gliedert.

Links der ersten Messdatenflussdarstellung 1 ist ein Eingang der Bitbreite  $k$  gezeigt, an dem die Messdatenworte  $y(1)$ ,  $y(2)$ , ...,  $y(n)$  bspw. von einem getesteten integrierten Schaltkreis jeweils getaktet ausgegeben werden. Daher sind die Messdatenworte mit den höheren Indizes in Figur 1 weiter links angeordnet. Ein solcher getesteter integrierter Schaltkreis wird nachfolgend auch Device Under Test/DUT bezeichnet.

Rechts der ersten Messdatenflussdarstellung 1 ist ein Ausgang der Bitbreite  $k$  dargestellt, über den die Messdaten einem Schieberegister 11 zugeleitet werden. Bei dem Schieberegister 11 handelt es sich um ein Schieberegister mit einer Multiple-Input-Linear-Feedback-Shift-Register-Architektur bzw. mit einer MILFSR-Architektur. Durch das Schieberegister 11 sollen alle  $n$  Datenworte in einem einzigen Datenwort der Länge  $k$  Bit komprimiert werden. Die Vorschrift für die Abbildung von  $n$  Datenworten  $y(1)$ ,  $y(2)$ , ...,  $y(n)$  in eine einzige Signatur ist in Gleichung (1) gegeben.

$$S(1 \rightarrow n) = S(y(1), y(2), \dots, y(n)) \quad (1)$$

Der Anfangszustand  $z(t_0)$  des Schieberegisters 11 mit der MILFSR-Architektur ist durch den Nullvektor 0 gegeben,  $z(t_0) = 0$ .

Figur 2 zeigt eine zweite Messdatenflussdarstellung 2 mit einem fehlerhaften Messdatenwort  $y'(i)$ .

Die zweite Messdatenflussdarstellung 2 unterscheidet sich von der ersten Messdatenflussdarstellung 1 dadurch, dass ihr Datenstrom  $T_n$  einen Fehler  $e(i)$  im  $i$ -ten Datenwort  $y'(i)$  des Datenstroms  $T_n$  aufweist.

Wenn ein solcher Fehler  $e(i)$  im Datenwort  $y'(i)$  des Datenstroms  $T_n$  vorliegt, ist die Vorschrift für die Abbildung von  $n$  Datenworten  $y(1), y(2), \dots, y(n)$  durch die Signatur des Schieberegisters 11 durch Gleichung (2) gegeben.

$$S(1 \rightarrow n, i, e) = S(y(1), y(2), \dots, e(i), \dots, y(n)) \quad (2)$$

Diese Signatur wird mit dem Schieberegister 11 nun aus den Datenworten  $y(1), y(2), \dots, e(i), \dots, y(n)$  erzeugt. Das Schieberegister 11 weist die bereits mit Bezug auf Figur 1 erwähnte MILFSR-Architektur auf.

Durch geschickten Vergleich der Signaturen  $S(y(1), y(2), \dots, e(i), \dots, y(n))$  in Gleichung (1) und  $S(y(1), y(2), \dots, e(i), \dots, y(n))$  in Gleichung (2) kann auf die Position  $i$  des fehlerhaften Datenworts  $y'(i)$  und auf den Fehler  $e(i)$ , d. h. auf alle seine fehlerhaften Bits  $e(i)$  zurückgerechnet werden.

20

Enthält eine bekannte digitale Testantwort eines DUT genau einen Fehler  $e(i)$  im  $i$ -ten Datenwort des Datenstroms, so können mit Hilfe der aus einer MILFSR Architektur berechneten Abbildung  $S(1 \rightarrow n, i, e)$  das Datenwort  $y'(i)$  und alle fehlerhaften Bits seines Fehlers  $e(i)$  gewonnen werden. Dies geschieht zu einem Zeitpunkt  $t > n$ , an dem die Datenworte  $y(1), y(2), \dots, y(n)$  des Datenstroms  $T_n$  nicht mehr zur Verfügung stehen bzw. nicht mehr benötigt werden.

Gemäß der vorliegenden Erfindung brauchen für den Test, für die Diagnose und für die Fehlerkorrektur eines Fehlers in einem beliebigen Datenwort des Datenstroms  $T_n$  aus  $n$   $k$ -bit Daten-



worten nur 2 Signaturen, d.h. 2 k-bit Datenworte ausgewertet zu werden.

Figur 3 zeigt eine schematische Darstellung einer ersten linearen Automatenschaltung L1.

Die erste lineare Automatenschaltung L1 umfasst vier in Figur 3 rechteckig dargestellte und jeweils nacheinander angeordnete Zustände  $z_1$ ,  $z_2$ ,  $z_3$  und  $z_4$ , die in Speicherelementen, bspw. D-Flip Flops gespeichert sind. Vor dem ersten Zustand  $z_1$  und jeweils zwischen den Zuständen  $z_2$ ,  $z_3$  und  $z_4$  sind vier exklusive Oder-Gatter  $XOR_1$ ,  $XOR_2$ ,  $XOR_3$  und  $XOR_4$  vorgesehen, deren erste Eingänge jeweils von vier Messdatenleitungen  $y_1$ ,  $y_2$ ,  $y_3$  und  $y_4$  gebildet werden und deren Ausgänge mit dem jeweils nachfolgenden Zuständen  $z_1$ ,  $z_2$ ,  $z_3$  und  $z_4$  verbunden sind.

Der zweite Eingang des zweiten Oder-Gatters  $XOR_2$  ist mit dem Ausgang des ersten Zustands  $z_1$  verbunden. Der zweite Eingang des dritten Oder-Gatters  $XOR_3$  wird von dem Ausgang des zweiten Zustands  $z_2$  gebildet. Den zweiten Eingang des vierten Oder-Gatters  $XOR_4$  bildet der Ausgang des dritten Zustands  $z_3$ .

Am Ausgang des vierten Zustands  $z_4$  setzt eine erste Ausgangsleitung 30 an, die sich in eine erste Rückkopplungsleitung 31 und in eine zweite Rückkopplungsleitung 32 gabelt. Die erste Rückkopplungsleitung 31 bildet den dritten Eingang des zweiten XOR-Gatters  $XOR_2$ , und die zweite Rückkopplungsleitung 32 bildet den zweiten Eingang des ersten XOR-Gatters  $XOR_1$ .

Die erste lineare Automatenschaltung L1 ist dementsprechend als ein mit einem primitiven Polynom rückgekoppelten Schieberegisters mit vier Zuständen  $z_1$ ,  $z_2$ ,  $z_3$  und  $z_4$  ausgebildet.

Die Zustandsgleichungen der ersten linearen Automatenschaltung L1 werden aus den Zuständen  $z(t+1)$  zum Zeitpunkt  $t+1$  in Abhängigkeit der Zustände  $z(t)$  zum Zeitpunkt  $t$  berechnet. Für die erste lineare Automatenschaltung L1 gelten die Gleichungen (3) - (6):

$$z_1(t+1) = z_4(t) \oplus y_1(t) \quad (3)$$

$$z_2(t+1) = z_1(t) \oplus z_4(t) \oplus y_2(t) \quad (4)$$

$$z_3(t+1) = z_2(t) \oplus y_3(t) \quad (5)$$

$$z_4(t+1) = z_3(t) \oplus y_4(t) \quad (6)$$

Mit  $A$  als Zustandsmatrix ergibt sich:

$$z(t+1) = Az(t) \oplus y(t) \quad (7)$$

15

mit

$$A = \begin{bmatrix} 0001 \\ 1001 \\ 0100 \\ 0010 \end{bmatrix}, \quad z(t+1) = \begin{bmatrix} z_1(t+1) \\ z_2(t+1) \\ z_3(t+1) \\ z_4(t+1) \end{bmatrix}, \quad z(t) = \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \end{bmatrix}, \quad y(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \\ y_4(t) \end{bmatrix}$$

20 Das autonome Verhalten der ersten linearen Automatenschaltung L1 berechnet sich aus Gleichung (7) mit  $y(t)=0$  zu

$$z(t+1) = Az(t) \quad (8)$$

25 Die Gleichungen (3) - (8) werden auch als Nachfolger-Gleichungen bezeichnet.

Aus den Gleichungen (3), (5) und (6) folgt mit  $y_i(t) = 0$  für  $i \in \{1,2,3,4\}$ :

30

$$z_4(t) = z_1(t+1) \quad (9)$$

$$z_2(t) = z_3(t+1) \quad (10)$$

$$z_3(t) = z_4(t+1) \quad (11)$$

5 Aus Gleichung (9), eingesetzt in Gleichung (4) folgt:

$$z_2(t+1) = z_1(t+1) \oplus z_1(t)$$

und

10

$$z_1(t+1) \oplus z_2(t+1) = z_1(t+1) \oplus z_1(t+1) \oplus z_1(t)$$

Daraus ergibt sich:

$$15 \quad z_1(t) = z_1(t+1) \oplus z_2(t+1) \quad (12)$$

Die Gleichungen (9) - (12) werden als Vorgänger-Gleichungen der linearen Automatenschaltung L1 für die Eingabe 0 oder auch als autonome Vorgänger-Gleichungen der linearen Automaten-  
20 schaltung L1 bezeichnet.

In Matrizenschreibweise erhält man analog zu den Nachfolger-Gleichungen in (8) aus (9)-(12) die Vorgänger-Gleichungen

$$25 \quad z(t) = A^{-1} z(t+1) \quad (13)$$

$$\text{mit } A^{-1} = \begin{bmatrix} 1100 \\ 0010 \\ 0001 \\ 1000 \end{bmatrix}$$

$A^{-1}$  ist die Invertierte zu A. Die Multiplikation aus A und aus  
30  $A^{-1}$  ergibt die Einheitsmatrix E.

$$A^{-1} A = E \text{ mit } E = \begin{bmatrix} 1000 \\ 0100 \\ 0010 \\ 0001 \end{bmatrix} \quad (14)$$

Figur 4 zeigt eine schematische Darstellung einer zweiten li-  
5 nearen Automatenschaltung L2.

Bezüglich der Signalrichtung ist die zweite lineare Automaten-  
schaltung L2 spiegelbildlich zur ersten linearen Automaten-  
schaltung L1 dargestellt.

10

Die zweite lineare Automatenschaltung L2 umfasst wie die erste  
lineare Automatenschaltung L1 vier Zustände  $z_1$ ,  $z_2$ ,  $z_3$  und  $z_4$ ,  
vier exklusive Oder-Gatter  $XOR_1$ ,  $XOR_2$ ,  $XOR_3$  und  $XOR_4$  sowie vier  
Messdatenleitungen  $y_1$ ,  $y_2$ ,  $y_3$  und  $y_4$ , welche jeweils die ersten  
15 Eingänge der exklusiven Oder-Gatter  $XOR_1$ ,  $XOR_2$ ,  $XOR_3$  und  $XOR_4$   
bilden.

: 20

Die exklusiven Oder-Gatter  $XOR_1$ ,  $XOR_2$ ,  $XOR_3$  und  $XOR_4$  befinden  
sich vor dem vierten Zustand  $z_4$  und jeweils zwischen den Zu-  
ständen  $z_1$ ,  $z_2$  und  $z_3$ . Die Ausgänge der exklusiven Oder-Gatter  
 $XOR_1$ ,  $XOR_2$ ,  $XOR_3$  und  $XOR_4$  sind mit dem jeweils nachfolgenden  
Zuständen  $z_1$ ,  $z_2$ ,  $z_3$  und  $z_4$  verbunden.

: 25

Der Ausgang des ersten Zustands  $z_1$  bildet eine zweite Aus-  
gangsleitung 40, die sich in eine dritte Rückkopplungsleitung  
41 und in eine vierte Rückkopplungsleitung 42 aufteilt. Die  
dritte Rückkopplungsleitung 41 fließt in das erste exklusive  
Oder-Gatter  $XOR_1$  und die vierte Rückkopplungsleitung 42 in das  
vierte exklusive Oder-Gatter  $XOR_4$  ein.

: 30

Wie die erste lineare Automatenschaltung L1 ist auch die zweite lineare Automatenschaltung L2 als ein mit einem primitiven Polynom rückgekoppeltes Schieberegisters mit vier Zuständen  $z_1, z_2, z_3$  und  $z_4$  ausgebildet.

5

Die Zustandsgleichung der zweiten linearen Automatenschaltung L2 ist in Gleichung (15) gegeben.

$$z(t+1) = B z(t) \oplus y(t) \quad (15)$$

10

mit

$$B = \begin{bmatrix} 1100 \\ 0010 \\ 0001 \\ 1000 \end{bmatrix} = A^{-1} \quad (16)$$

B ist die Invertierte von A.

15

Für einen angenommenen Fehler  $e(i)$  in einem k-bit Datenwort  $y'(i)$  des Datenstroms  $T_n$  aus Figur 2 erhält man aus der Abbildungsvorschrift gemäß Gleichung (2)

$$S(1 \rightarrow n, i, e) = S(y(1), y(2), \dots, y(i) \oplus e(i), \dots, y(n)) \quad (17)$$

20

und durch Anwendung des Superpositionsprinzips bei linearen Automaten

$$S(1 \rightarrow n, i, e) = S(y(1), y(2), \dots, y(n)) \oplus S(0, 0, \dots, e(i), \dots, 0)$$

25

bzw.

$$S(1 \rightarrow n, i, e) = S(1 \rightarrow n) \oplus S(0, 0, \dots, e(i), \dots, 0)$$

30

die Differenz

$$\Delta S1(i,e) = S(1 \rightarrow n,i,e) \oplus S(1 \rightarrow n) \quad (18)$$

5 mit

$$\Delta S1(i,e) = S(0,0,\dots,e(i),\dots,0) \quad (19)$$

für

10

$$y(t)=0, \text{ für } t \neq i \text{ und } y(t)=e(i), \text{ für } t=i$$

Mit Gleichung (19) in Gleichung (18) folgt

$$15 \quad \Delta S1(i,e) = S((n-i) \rightarrow n,i,e) \oplus S((n-i) \rightarrow n) \quad (20)$$

Zum Zeitpunkt  $t = i-1$  ist noch kein Fehler  $e(i)$  im Datenwort  $y'(i)$  aufgetreten; erst ab dem Zeitpunkt  $t=i$  tritt dieser Fehler  $e(i)$  in einem  $k$ -bit Datenwort auf.

20

Für die Bildung der Signatur aus  $n$  Datenworten  $y(1), y(2), \dots, y(n)$  mit genau einem angenommenen Fehler  $e(i)$  in einem Datenwort  $y'(i)$  erhalten wir aus den Gleichungen (20) und (7)

$$25 \quad \Delta S1(i,e) = A^{(n-1)}e(i); i \in \{1,2,\dots,n\} \quad (21)$$

Gleichung (21) ist die Differenz von zwei MILFSR-Signaturen, nämlich von der Signatur eines 1-Wort fehlerhaften Datenstroms  $T_n$  am Ort  $i$  und von der Signatur eines fehlerfreien Datenstroms  $T_n$ . Die Signaturdifferenz  $\Delta S1(i,e)$  wird gebildet, indem  
 30 das fehlerhafte Datenwort  $y'(i)$  den Anfangszustand der MILFSR



Architektur definiert und dann  $(n-i)$ -mal in der MILFSR Architektur bei Eingabe von lauter 0-Vektoren akkumuliert wird.

Formt man nun Gleichung (21) mit Gleichung (16) um

5

$$e(i) = B^{(n-i)} \Delta S1(i, e) \quad (22)$$

so wird erkennbar, dass das Datenwort  $e(i)$  als neue Signatur berechnet werden kann, indem der Anfangszustands der zweiten  
10 linearen Automatenschaltung L2 auf  $\Delta S1(i, e)$  gesetzt und danach  $(n-i)$ -mal akkumuliert wird.

Erfindungsgemäß können invertierte MILFSR Architekturen also zur Berechnung von 1-Wort Fehlern  $e$  in Datenströmen  $T_n$  verwenden  
15 werden, ohne dass es notwendig ist, diese Datenströme zum Zeitpunkt der Berechnung von  $e(i)$  noch zur Verfügung zu haben.

Aus der der Gleichung (20) entsprechenden Gleichung

$$20 \quad \Delta S2(i, e) = B^{(n-i)} e(i) \quad (23)$$

für die zweite lineare Automatenschaltung L2 erhalten wir mit Gleichung (16) analog zu Gleichung (20) die folgende Gleichung  
(24). Für einen angenommenen 1-Wort Fehler  $e(i)$  erhält man aus  
25 Gleichung (21) für die erste lineare Automatenschaltung L1

$$e(i) = A^{(n-i)} \Delta S2(i, e) \quad (24)$$

Durch Gleichsetzen von Gleichung (22) und (24) ergibt sich

30

$$A^{(n-i)} \Delta S2(i, e) = B^{(n-i)} \Delta S1(i, e) \quad (25)$$

Wenn also der Anfangszustand der ersten linearen Automaten-  
 schaltung L1 auf  $\Delta S2(i,e)$  bzw. wenn der Anfangszustand der  
 zweiten linearen Automaten-schaltung L2 auf  $\Delta S1(i,e)$  gesetzt  
 wird und wenn dieser Wert  $(n-i)$ -mal akkumuliert wird, so er-  
 5 gibt sich der Fehler  $e(i)$  bei beiden linearen Automaten-schal-  
 tungen L1 und L2 als neue Signatur bzw. als neuer Zustand.

Der Wert für die Position  $i$  des fehlerhaften Datenworts  $y'(i)$   
 muss dabei zwischen 1 und  $n$  liegen.

10

Ergibt sich für keinen der Werte  $i$ ,  $1 \leq i \leq n$  in Gleichung (25)  
 die Gleichheit der Ausdrücke  $A^{(n-i)}\Delta S2(i,e)$  und  $B^{(n-i)}\Delta S1(i,e)$ , so  
 liegt kein Fehler  $e(i)$  vor, der nur die Position  $i$  des Daten-  
 worts verfälscht, sondern ein Fehler, der mindestens zwei un-  
 15 terschiedliche Positionen im Datenstrom verfälscht hat.

Ergibt sich  $\Delta S1 \neq 0$ , dann kann man durch Zurücksetzen der line-  
 aren Automaten-schaltungen L1 und L2 auf  $\Delta S2$  bzw.  $\Delta S1$  und  
 durch Vergleich der in jedem Schritt akkumulierten Signaturen  
 20 die Position  $i$  des fehlerhaften Datenworts  $y'(i)$  und den Feh-  
 ler  $e(i)$  nach  $n-i$  Schritten an beiden Signaturen bzw. Zustän-  
 den detektieren. Die Signaturen beider linearer Automaten-  
 schaltungen L1 und L2 sind in diesem Fall gleich

$$25 \quad z^1(n-i) = z^2(n-i) \quad (26),$$

wobei  $z^1$  und  $z^2$   $L$ -dimensionale Zustandsvektoren der linearen  
 Automaten-schaltungen L1 und L2 sind, und wobei  $z^1(n-i)$  und  
 $z^2(n-i)$  die Zustände der beiden linearen Automaten-schaltungen  
 30 L1 und L2 nach  $n-i$  Takten sind.

Bei serielllem Dateninput gilt für die beiden linearen Automaten-schaltungen L1 und L2

$$y_i(t)=0 \text{ für } i \neq 1 \text{ (L1)} \quad (27)$$

$$5 \quad y_i(t)=0 \text{ für } i \neq k \text{ (L2)} \quad (28)$$

Dabei steht k für die Anzahl der Bits eines Datenwortes  $y(i)$  aus dem Datenstrom  $T_n$ .

- 10 Der Vergleich auf Identität der in dieser MILFSR Architektur zurückgerechneten Signaturen ergibt für einen angenommenen Fehler e im Datenwort i nach n-1 Schritten

$$z^1(n-i) = z^2(L+1-n+i) \quad (29)$$

15

Figur 5 zeigt ein Ablaufschema 5 zum Bestimmen von Signaturunterschieden  $\Delta S1$  und  $\Delta S2$  aus einem Messdatenstrom  $T_n$ .

An der oberen Seite des Ablaufschemas 5 befindet sich eine  
20 Messdatenwortleitung 51, an der ein Datenstrom  $T_n$  aus n aufeinanderfolgenden Datenwörtern  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  mit jeweils gleicher Datenwortlänge von k Bit anliegt. Dabei stellt das Datenwort  $y'(i)$  ein fehlerhaftes Datenwort dar.

25

Die Messdatenwortleitung 51 gabelt sich in zwei Äste, von denen der linke einen ersten Kodierer C1, die erste lineare Automaten-schaltung L1, L-exklusive Oder-Gatter  $XOR_{L1}$ , die in Form eines XOR-Gatters der Wortbreite L zusammengefasst sind,  
30 sowie einen ersten Ausgang 52 aufweist, an dem der Signaturunterschied  $\Delta S1$  ausgegeben wird. Der rechte Ast umfasst einen zweiten Kodierer C2, die zweite lineare Automaten-schaltung L2,

L-exklusive Oder-Gatter  $XOR_{L2}$ , die ebenfalls in Form eines XOR-Gatters der Wortbreite L zusammengefasst sind, sowie einen zweiten Ausgang 53, an dem der Signaturunterschied  $\Delta S2$  ausgegeben wird.

5

Die jeweils L-ersten Eingänge der L-exklusiven Oder-Gatter  $XOR_{L1}$  werden von den Ausgängen der ersten linearen Automaten-schaltung L1, und die ersten Eingänge der L-exklusiven Oder-Gatter  $XOR_{L2}$  werden von den Ausgängen der zweiten linearen Au-  
10 tomatenschaltung L2 gebildet.

An den zweiten Eingängen der L-exklusiven Oder-Gatter  $XOR_{L1}$  und  $XOR_{L2}$  werden durch in Figur 5 nicht gezeigte digitale Schaltungen Gutsignaturen der beiden linearen Automaten-schal-  
15 tungen L1 und L2 bereitgestellt. Diese Gutsignaturen können für einen bekannten Datenstrom  $T_n$  bestimmt werden. Dies ist dem Fachmann bekannt.

Der Messdatenstrom  $T_n$  wird über den ersten Kodierer C1 in die  
20 erste lineare Automaten-schaltung L1 und parallel dazu über den zweiten Kodierer C2 in die zweite lineare Automaten-schaltung L2 eingegeben.

Durch die Kodierer C1 und C2 wird der jeweils anliegende Da-  
25 tenstrom  $T_n$  einheitlich transformiert. Dadurch ist es möglich, auch Datenwörter  $y(1), \dots, y(n)$  verschiedener Datenströme  $T_n$  miteinander zu vergleichen.

Durch die linearen Automaten-schaltungen L1 und L2 werden die  
30 durch die Kodierer C1 und C2 kodierte Datenwörter  $u^1(1), \dots, u^1(n)$  bzw.  $u^2(1), \dots, u^2(n)$  zu Signaturen S1 und S2 kompaktiert.

Die so bestimmten tatsächlichen Signaturen  $S_1$  und  $S_2$  werden danach in den exklusiven Oder-Gattern  $XOR_{L1}$  und  $XOR_{L2}$  mit den bekannten Gutsignaturen verglichen. Die jeweils L-exklusiven Oder-Gatter  $XOR_{L1}$  und  $XOR_{L2}$  stellen die Signaturunterschiede  $\Delta S_1$  und  $\Delta S_2$  an den Ausgängen 52 und 53 bereit.

Gemäß der Erfindung kann mithilfe dieser Signaturunterschiede  $\Delta S_1$  und  $\Delta S_2$  festgestellt werden, ob bei dem jeweils betrachteten Datenstrom  $T_n$  Fehler aufgetreten sind.

10

Wenn der Datenstrom  $T_n$  genau ein fehlerhaftes Datenwort  $y'(i)$  aufweist, kann sowohl die Position  $i$  des fehlerhaften Datenworts als auch der Fehler  $e(i)$  des fehlerhaften Datenworts im Datenstrom direkt bestimmt werden.

15

Nachfolgend ist die Funktion des ersten Kodierers  $C_1$  genau beschrieben.

Der Kodierer  $C_1$  kodiert für  $i=1, \dots, n$  das Datenwort  $y(i)$  mit der Datenwortlänge von  $k$  Bit in ein kodiertes Datenwort  $u^1(i)$ ,  $u^1(i) = \text{Cod1}(y(i))$  der Wortbreite von  $K_1$  Bit. Dabei bezeichnet  $\text{Cod1}$  die Kodierungsfunktion des ersten Kodierers  $C_1$ .

Ist  $y'(i)$  die komponentenweise XOR-Summe von zwei  $k$  Bit breiten Datenwörtern  $y(i)$  und  $e(i)$ , nämlich

$$y'(i) = y(i) \oplus e(i)$$

dann soll für die Kodierungsfunktion  $\text{Cod1}$

30

$$\begin{aligned} u^{1'}(i) &= \text{Cod1}(y'(i)) = \text{Cod1}(y(i) \oplus e(i)) \\ &= \text{Cod1}(y(i)) \oplus f_1(e(i)) = u^1(i) \oplus f_1(e(i)) \end{aligned}$$

gelten, so dass es eine Funktion  $f_1^{-1}$  mit

$$f_1^{-1}(f_1(e)) = e$$

- 5 für alle binären Datenworte  $e$  der Wortbreite  $k$ , die als Fehler möglich sind, gilt.

Realisiert der Kodierer  $C_1$  einen linearen Blockcode mit  $k$  Informationsstellen und mit  $(K_1-k)$  Kontrollstellen, dann gilt

: 10

$$\begin{aligned} \text{Cod}_1(y'(i)) &= \text{Cod}_1(y(i) \oplus e(i)) \\ &= \text{Cod}_1(y(i)) \oplus \text{Cod}_1(e(i)) \end{aligned}$$

und es gilt

: 15

$$f_1 = \text{Cod}_1.$$

Lineare Blockcodes sind bspw. in Dokument [1] beschrieben.

: 20

Sind z.B.  $k = 3$  und  $K_1 = 5$  und gilt

$$\begin{aligned} C_1(y(i)) &= C_1([y_1(i), y_2(i), y_3(i)]^T) \\ &= [y_1(i), y_2(i), y_3(i), 1, 1]^T, \end{aligned}$$

: 25

so gilt für  $y'(i) = y(i) \oplus e(i)$ :

$$\begin{aligned} \text{Cod}_1(y'(i)) &= \text{Cod}_1(y(i) \oplus e(i)) \\ &= \text{Cod}_1(y(i)) \oplus f(e(i)) \end{aligned}$$

: 30

mit

$$f_1(e(i)) = f_1([e_1(i), e_2(i), e_3(i)]^T)$$



$$= [e_1(i), e_2(i), e_3(i), 0, 0]$$

und

$$5 \quad f_1^{-1}([e_1(i), e_2(i), e_3(i), 0, 0]^T) = e_1(i), e_2(i), e_3(i).$$

Nachfolgend ist die Funktion der ersten linearen Automaten-schaltung L1 im einzelnen beschrieben.

10 Bei der nachfolgenden Beschreibung kann die Zustandsmatrix der linearen Automaten-schaltungen L1 und L2 eine beliebige Größe annehmen und ist nicht auf  $n=4$  festgelegt, wie zuvor beschrieben. Ein weiterer Unterschied der nachfolgenden Beschreibung liegt darin, dass es sich bei den Eingabewerten der linearen  
15 Automaten-schaltungen L1 und L2 um kodierte Datenwörter  $u^1(1), \dots, u^1(n)$  bzw.  $u^2(1), \dots, u^2(n)$  handelt, die vom ersten Kodierer C1 bzw. vom zweiten Kodierer C2 aus den Datenwörtern  $y(1), \dots, y(n)$  kodiert worden sind.

20 Die erste lineare Automaten-schaltung L1 ist über dem Körper  $GF(2)$  ausgebildet und weist einen  $K_1$ -dimensionalen Zustandsvektor  $z^1(t)$  auf. Dabei gilt  $K_1 \geq k$ .

Die erste lineare Automaten-schaltung L1 ist allgemein durch  
25 die Gleichung

$$z^1(t+1) = A \cdot z^1(t) \oplus u^1(t) \tag{I}$$

beschrieben. Dabei sind  $z^1(t)$  und  $z^1(t+1)$   $K_1$ -dimensionale bi-  
30 näre Zustandsvektoren zum diskreten Zeitpunkt  $t$  und  $t+1$ .  $u^1(t)$  ist das zum Zeitpunkt  $t$  eingegebene kodierte Datenwort.

A ist eine eindeutig umkehrbare binäre ( $K_1 \times K_1$ )-Matrix.

Die Addition und die Multiplikation der binären Werte in Gleichung (I) erfolgen modulo 2.

5

Nach Eingabe des Datenstroms  $T_n$  aus den  $n$  Datenwörtern  $y(1), \dots, y(n)$  in den ersten Kodierer  $C_1$  und nach Eingabe der kodierten Datenwörter  $u^1(1), \dots, u^1(n)$  in die erste lineare Automatenschaltung  $L_1$  geht die erste lineare Automatenschaltung  $L_1$  von einem Anfangszustand  $z^1(1)$  in den Zustand  $z^1(n+1)$  über. Dabei gilt:

$$z^1(n+1) = A^n z^1(1) \oplus \sum_{j=1}^n A^{n-j} u^1(j)$$

15 Dabei wird

$$z^1(n+1) = S(L_1, u^1(1), \dots, u^1(n))$$

mit  $u^1(i) = \text{Cod}_1(y(i))$  als Signatur der Datenfolge  $y(1), \dots, y(n)$  der ersten linearen Automatenschaltung  $L_1$  bezeichnet.

Der Anfangszustand  $z^1(1)$  soll nachfolgend gleich dem  $K_1$ -dimensionalen Nullvektor 0 sein. Für die Durchführung des erfindungsgemäßen Verfahrens ist es nicht notwendig, dass der Anfangszustand  $z^1(1)$  als Nullvektor ausgebildet ist. Diese Annahme wird hier getroffen, um die nachfolgenden Rechnungen zu vereinfachen.

30 Liegt nun ein Fehler  $e(i)$  im  $i$ -ten Datenwort des Datenstroms vor, dann wird zum  $i$ -ten Zeitpunkt anstatt des richtigen Datenwortes  $y(i)$  das fehlerhafte Datenwort

$$y'(i) = y(i) \oplus e(i)$$

in den ersten Kodierer C1 eingegeben. Dabei ist der Fehler  
 5  $e(i)$  ein  $k$ -dimensionaler Binärvektor. Diejenigen Komponenten  
 dieses Binärvektors, die den Wert Eins annehmen, bezeichnen  
 die Positionen der fehlerhaften Bits in dem  $i$ -ten Datenwort.

Ist nur das  $i$ -te Datenwort fehlerhaft und sind alle anderen  
 10 Datenwörter korrekt, so gilt mit  $z^1(1) = 0$  bei Eingabe des nur  
 im  $i$ -ten Datenwort fehlerhaften Datenstroms in den ersten Ko-  
 dierer C1 mit

$$\begin{aligned} u^{1'}(i) &= \text{Cod1}(y'(i)) \\ 15 \quad &= \text{Cod1}(y(i) \oplus e(i)) \\ &= \text{Cod1}(u(i)) \oplus f_1(e(i)) \\ &= u^1(i) \oplus f_1(e(i)) \end{aligned}$$

$$\begin{aligned} z^{1'}(n+1) &= \sum_{j=1}^n A^{n-j} u^1(j) \oplus A^{n-1} \cdot f_1(e(i)) & \text{(II)} \\ 20 \quad &= z^1(n+1) \oplus A^{n-1} f_1(e(i)) & \text{(III)} \end{aligned}$$

und damit

$$\begin{aligned} A^{n-i} \cdot f_1(e(i)) &= z^{1'}(n+1) \oplus z^1(n+1) \\ 25 \quad &= S(L1, u^1(1), \dots, u^{1'}(i), \dots, u^1(n)) \\ &\quad \oplus S(L1, u^1(1), \dots, u^1(i), \dots, u^1(n)) \\ &= \Delta S1 & \text{(IV)} \end{aligned}$$

Für den Fall, dass  $f(e(i)) = \text{Cod1}(e(i))$  ist, erhält man  
 30

$$A^{n-i} \text{Cod1}(e(i)) = \Delta S1 \quad \text{(V)}$$

Für den Fall, dass  $\text{Cod1}(y(i)) = y(i)$ ,  $i=1, \dots, n$  gilt, ergibt sich

$$5 \quad A^{n-i} \cdot e(i) = \Delta S1 \quad (\text{VI}).$$

Nachfolgend ist die Funktion des zweiten Kodierers C2 genau beschrieben.

10

Der zweite Kodierer C2 kodiert für  $i=1, \dots, n$  das Datenwort  $y(i)$  mit der Datenwortlänge von  $k$  Bit in das kodierte Datenwort  $u^2(i)$ ,  $u^2(i) = \text{Cod2}(y(i))$ . Das kodierte Datenwort  $u^2(i)$  hat eine Wortbreite von  $K2$  Bit. Dabei bezeichnet  $\text{Cod2}$  die Kodierungsfunktion des zweiten Kodierers C2.

15

Ist  $y'(i)$  die komponentenweise XOR-Summe von zwei  $k$  Bit breiten Wörtern  $y(i)$  und  $e(i)$ .

$$20 \quad y'(i) = y(i) \oplus e(i),$$

dann soll für die Kodierungsfunktion  $\text{Cod2}$

$$\begin{aligned} 25 \quad u^{2'}(i) &= \text{Cod2}(y'(i)) \\ &= \text{Cod2}(y(i) \oplus e(i)) \\ &= \text{Cod2}(y(i)) \oplus f_2(e(i)) \\ &= u^2(i) \oplus f_2(e(i)) \end{aligned}$$

gelten, so dass es eine Funktion  $f_2^{-1}$  mit  $f_2^{-1}(f_2(e)) = e$  für  
30 alle binären Datenworte  $e$  der Wortbreite  $k$ , die als Fehler in Betracht kommen können, gibt.

Nachfolgend ist die Funktion der zweiten linearen Automaten-schaltung L2 im einzelnen beschrieben.

Die zweite lineare Automatenschaltung L2 ist über dem Körper  
5 GF(2) ausgebildet und weist einen K2-dimensionalen Zustands-  
vektor  $z^2(t)$  auf. Dabei gilt  $K2 \geq k$ .

Die zweite lineare Automatenschaltung L2 ist durch die Gleichung

10

$$z^2(t+1) = B \cdot z^2(t) u^2(t) \quad (\text{VII})$$

beschrieben, wobei  $z^2(t)$  und  $z^2(t+1)$  K2-dimensionale binäre Zustandsvektoren zu den diskreten Zeitpunkten t und t+1 sind.

15

$u^2(t)$  ist das zum Zeitpunkt t eingegebene kodierte Datenwort. B mit  $B \neq A$  ist eine eindeutig umkehrbare binäre (K2xK2)-Matrix. Die Addition und Multiplikation in (VII) erfolgt modulo 2.

20

Nach Eingabe des Datenstroms  $T_n$  aus den n Datenwörtern  $y(1), \dots, y(n)$  in den zweiten Kodierer C2 und nach Eingabe der kodierten Datenwörter  $u^2(1), \dots, u^2(n)$  in die zweite lineare Automatenschaltung L2 geht die zweite lineare Automaten-schaltung L2 aus ihrem Anfangszustand  $z^2(1)$  in den Zustand

: 25

$z^2(n+1)$

$$z^2(n+1) = B^n z^2(1) \oplus \sum_{j=1}^n B^{n-j} u^2(j) \quad (\text{VIII})$$

über. Dabei ist

: 30

$$z^2(n+1) = S(L2, u^2(1), \dots, u^2(n))$$

mit  $u^2(i) = \text{Cod2}(y(i))$  als die Signatur des Datenstromes  $y(1), \dots, y(n)$  in der zweiten linearen Automatenschaltung L2 bezeichnet.

- 5 Der K1-dimensionale Nullvektor 0 bildet nachfolgend den Anfangszustand  $z^2(1)$ .

Liegt nun ein Fehler  $e(i)$  im  $i$ -ten Datenwort des Datenstromes vor, dann wird im  $i$ -ten Zeitpunkt anstatt des richtigen Datenwortes  $y(i)$  das fehlerhafte Datenwort  $y'(i) = y(i) \oplus e(i)$  in  
 10 den zweiten Kodierer Cod2 eingegeben.

Ist nur das  $i$ -te Datenwort fehlerhaft und sind alle anderen Datenwörter korrekt, so gilt mit  $z^2(1) = 0$ , mit

15

$$\begin{aligned} u^{2'}(i) &= \text{Cod2}(y(i) \oplus e(i)) \\ &= u^2(i) \oplus f_2(e(i)) \quad (\text{analog zu (III)}) \end{aligned}$$

20

$$\begin{aligned} z^{2'}(n+1) &= \sum_{i=1}^n B^{n-i} u^2(i) + B^{n-i} f_2(e(i)) \\ &= z^2(n+1) \oplus B^{n-i} f_2(e(i)) \quad (\text{IX}) \end{aligned}$$

und mit

25

$$\begin{aligned} z^{2'}(n+1) \oplus z^2(n+1) &= S(L2, u^2(1), \dots, u^{2'}(i), \dots, u^2(n)) \\ &\quad \oplus S(L2, u^2(1), \dots, u^2(i), \dots, u^2(n)) \\ &= \Delta S2 \end{aligned}$$

folgende Gleichung:

30

$$\Delta S2 = B^{n-i} f_2(e(i)) \quad (\text{X})$$

Für den Fall, dass  $f_2(e(i)) = \text{Cod2}(e(i))$  ist, gilt anstelle von (X)

$$5 \quad \Delta S2 = B^{n-i} \text{Cod2}(e(i)) \quad (\text{XI})$$

Für den Fall, dass  $\text{Cod2}(e(i)) = e(i)$  gilt, ergibt sich

$$\Delta S2 = B^{n-i} \cdot e(i) \quad (\text{XII})$$

10

Wenn die Ergebnisse der Verarbeitung des Datenstromes in den linearen Automaten-schaltungen L1 und L2 kombiniert werden, so erhält man aus den Gleichungen (IV) und (X)

$$15 \quad e(i) = f_1^{-1}(A^{i-n} \Delta S1) \quad (\text{XIV})$$

$$e(i) = f_2^{-1}(B^{i-n} \Delta S2) \quad (\text{XV})$$

woraus sich folgende Gleichung ergibt:

$$20 \quad f_1^{-1}(A^{i-n} \Delta S2) = f_2^{-1}(B^{i-n} \Delta S2) \quad (\text{XVI})$$

Mittels der Gleichung (XVI) kann  $i$ ,  $i \in \{1, \dots, n\}$ , also die Position  $i$  des fehlerhaften Datenworts  $y'(i)$  berechnet werden.

25 Der Wert für  $i$  aus Gleichung (XVI) kann bspw. iterativ dadurch berechnet werden, indem man für  $i = 1, 2, \dots$  den Wert der linken und rechten Seite der Gleichung (XVI) solange berechnet, bis beide Seiten übereinstimmen.

30 Wenn der Wert für  $i$  auf diese Weise bestimmt worden ist, dann kann mittels der Gleichung (XIV) auch der Wert für  $e(i)$ , also



die Position des fehlerhaften Datenworts im Datenstrom direkt bestimmt werden.

Figur 6 zeigt ein Ablaufdiagramm 6 zur Veranschaulichung des  
5 erfindungsgemäßen Verfahrens zum Feststellen eines Fehlers  $e(i)$  in einem Datenwort  $y'(i)$ .

Das erste Ablaufdiagramm 6 sieht insgesamt sieben Verfahrensschritte 61-67 vor, wobei die Verfahrensschritte 62 und 65 als  
10 Entscheidungsfelder und die restlichen Verfahrensschritte als Ausführungsfelder ausgebildet sind.

Im ersten Verfahrensschritt 61 werden die Signaturunterschiede  $\Delta S1$  und  $\Delta S2$  sowie ein Datenstrom  $T_n$  erzeugt, wie er bspw. in  
15 den Figuren 1 und 2 dargestellt ist. Dieser Datenstrom  $T_n$  kann aus einer Abfolge von Messdaten eines DUT gebildet sein.

Dieser Datenstrom  $T_n$  wird - wie mit Bezug auf Figur 5 ausführlich beschrieben - über den ersten Kodierer C1 der ersten linearen Automaten-schaltung L1 und gleichzeitig dazu über den  
20 zweiten Kodierer C1 der zweiten linearen Automaten-schaltung L2 zugeführt. Aus den von den linearen Automaten-schaltungen L1 und L2 erzeugten Signaturen und aus den bereitgestellten Gut-signaturen bestimmen die exklusiven Oder-Gatter  $XOR_{L1}$  und  $XOR_{L2}$   
25 die Signaturunterschiede  $\Delta S1$  und  $\Delta S2$  aus der Differenz der Gut- und Schlecht-Signaturen gemäß Gleichung (1) und Gleichung (2) und stellen die Signaturunterschiede  $\Delta S1$  und  $\Delta S2$  an den Ausgängen 52 und 53 bereit.

30 Im zweiten Verfahrensschritt 62 wird der am ersten Ausgang 52 anliegende Signaturunterschied  $\Delta S1$  mit Null verglichen. Ist der Signaturunterschied  $\Delta S1$  gleich Null, bedeutet dies, dass kein Fehler im Datenstrom  $T_n$  festgestellt werden konnte. In

diesem Fall wird mit der Überprüfung des nächsten Datenstroms  $T_n$  fortgefahren. Ist der Signaturunterschied  $\Delta S1$  ungleich Null, bedeutet dies, dass ein oder beliebig viele fehlerhafte Datenworte  $y(1), \dots, y(n)$  fehlerhaft sind. In diesem Fall  
5 wird mit dem Verfahrensschritt 63 fortgesetzt.

Im Verfahrensschritt 63 werden die Zustände  $z(t=n+1)$  der linearen Automatenschaltungen L1 und L2 jeweils auf die Werte der jeweils anderen Signaturunterschiede  $\Delta S1$  und  $\Delta S2$  zurück gesetzt.  
10

Im Verfahrensschritt 64 wird für  $y(t)=0$  zurückgerechnet, und zwar werden die linearen Automatenschaltungen L1 und L2 in  $(n-i)$  Schritten für  $y(t)=0$  akkumuliert und die Zustände nach  
15 Gleichung (26) auf Identität geprüft. Dieses Zurückrechnen ist dem Fachmann bekannt und braucht hier nicht weiter erläutert zu werden.

Im Verfahrensschritt 65 wird überprüft, ob eine Lösung existiert. Ist dies nicht der Fall, so ist mehr als ein Datenwort  $y(1), \dots, y(n)$  des Datenstroms  $T_n$  fehlerhaft.  
20

Wenn eine Lösung existiert, so werden das Datenwort  $y'(i)$  und der Fehler  $e(i)$  im Verfahrensschritt 66 berechnet.  
25

Zunächst wird dabei das Datenwort  $y'(i)$  aus der Anzahl der durchlaufenen Zyklen der linearen Automatenschaltung L1 und L2 bestimmt.

30 Danach wird der Fehler  $e(i)$  aus der Gleichung (30) berechnet:

$$e(i) = z(t) \text{ für } t=i, 1 \leq i \leq N \quad (30)$$

$z(t=i)$  ist der Zustand der beiden linearen Automaten-schaltungen L1 und L2 zum Zeitpunkt  $t=i$ .

- 5 In einem weiteren Ausführungsbeispiel können auch zwei serielle, also aufeinanderfolgende Datenströme  $T_n$  überprüft werden.

In diesem Fall wird der linke Ast der Messdatenwortleitung 51 zweimal hintereinander und der rechte Ast der Messdatenwort-  
10 leitung 51 gar nicht durchlaufen, wobei im zweiten Durchlauf der erste Kodierer C1 durch den zweiten Kodierer C2 ersetzt wird, falls sich der erste Kodierer C1 vom zweiten Kodierer C2 unterscheidet und die erste lineare Automaten-schaltung L1 mit der Zustandsmatrix A durch die zweite lineare Automaten-schal-  
15 tung L2 mit der Zustandsmatrix B,  $B \neq A$  ersetzt wird, was technisch bspw. einfach unter Verwendung von Multiplexern erfolgen kann.

Wird das Verfahren auf einem Computersystem realisiert, so ist  
20 ganz einfach derjenige Programmteil, der die erste lineare Automaten-schaltung L1 realisiert, im zweiten Durchlauf durch denjenigen Programmteil, der die zweite lineare Automaten-schaltung L2 realisiert, zu ersetzen. Ebenso ist, falls sich die Kodierer C1 und C2 voneinander unterscheiden, die Funktion  
25 Cod1 des ersten Kodierers C1 im ersten Durchlauf durch die Funktion Cod2 des zweiten Kodierers C2 im zweiten Durchlauf zu ersetzen.

In diesem Fall ist S1 die beim ersten Durchlauf errechnete  
30 Signatur und S2 die beim zweiten Durchlauf errechnete Signatur. Das Verfahren stimmt im wesentlichen mit dem im vorigen Ausführungsbeispiel beschriebenen Verfahren überein.

Die Ausführung des Verfahrens ist für einen Fachmann anhand der in dieser Patentschrift gegebenen Informationen ohne weiteres möglich.

5 Folgendes Dokument wurden im Rahmen dieser Patentschrift zitiert:

[1] Rohling H., Einführung in die Informations- und Codierungstheorie, Teubner Verlag, 1995

10

Um einen Fehler  $e(i)$  an der Position  $i$  des Datenstroms  $T_n$  eindeutig zu erkennen und zu diagnostizieren, ist es überraschenderweise nicht notwendig, dass das Rückkopplungspolynom primitiv oder irreduzibel ist. Es genügt hier tatsächlich, dass  
15 die Matrix  $B$  invers zur Matrix  $A$  ist.

So bestimmt sich der Wert der Position des Fehlers  $i$ , der natürlich im Datenstrom von 1 bis  $n$  auftreten muss, aus der Gleichung

20

$$\begin{aligned} f_1^{-1}(A^{i-n} \Delta S_1) &= f_2^{-1}(B^{i-n} \Delta S_2) \\ \text{und mit nun } B &= A^{-1} \text{ zu} \\ f_1^{-1}(A^{i-n} \Delta S_1) &= f_2^{-1}(A^{n-i} \Delta S_2). \end{aligned}$$

25 Und für  $e(i)$ , die fehlerhaften Bitpositionen im fehlerhaften  $i$ -ten Datenwort, ergibt sich

$$e(i) = f_1^{-1}(A^{i-n} \Delta S_1).$$

30 Besitzt die Matrix  $A$  eine inverse Matrix  $A^{-1}$ , so gilt  $A^k e(i) \neq \underline{0}$  für  $e(i) \neq \underline{0}$  für jedes  $k$ . Damit, und weil für lineare Schieberegister eine lineares Superpositionsprinzip gilt, kann

ein einzelner Fehler  $e(i)$ , der verschiedene Bits an der Position  $i$  verfälscht hat, niemals maskiert werden. Dadurch, dass die Signatur darüber hinaus noch in zwei unterschiedlichen Multi-Input Signaturregistern mit zueinander inversen Matizen  $A$  und  $B = A^{-1}$  bereitgestellt wird, ist es sogar möglich, die  
 5 Position  $i$  und den Wert  $e(i)$  exakt zu bestimmen.

Eine Maskierung von Fehlern in einer Signatur, die der Gutsignatur gleich ist, kann nur mit einer verschwindend kleinen  
 10 Wahrscheinlichkeit vorkommen, wenn mehrere Fehler an mehreren Bitpositionen auftreten, die sich zufällig in ihrer Signatur aufheben. Bei einer einzigen fehlerhaften Position, ist dies, wie ausgeführt, nicht möglich.

15 Die Wahrscheinlichkeit, dass beim Vorliegen von mehren Fehlern die Gleichung

$$f_1^{-1}(A^{i-n} \Delta S_1) = f_2^{-1}(A^{n-i} \Delta S_2)$$

20 für  $i$  mit  $1 \leq i \leq n$  eine Lösung hat, ist vernachlässigbar klein und kommt es vor, dass keine Lösung für  $i$  gefunden werden kann, bedeutet das, dass kein einzelner Fehler  $e(i)$ , der nur fehlerhafte Bits an der Position  $i$  im Datenstrom aufweist, vorliegt. Die Eigenschaft, dass das Rückführungspolynom primitiv sein soll, ist eine vorteilhafte Ausgestaltung, für die  
 25 die an sich schon praktisch vernachlässigbar kleine Fehlermaskierungswahrscheinlichkeit für Fehler, die mehrere Positionen im Datenstrom  $T_n$  betreffen, ganz besonders klein ist.

## Patentansprüche

1. Auswerteschaltung zum Feststellen und/oder zum Lokalisieren fehlerhafter Datenwörter in einem Datenstrom  $T_n$  mit den  
5 folgenden Merkmalen:

- eine erste lineare Automatenschaltung (L1) sowie eine parallel geschaltete zweite lineare Automatenschaltung (L2) mit jeweils einer Menge von Zuständen  $z(t)$ ,
- die erste lineare Automatenschaltung (L1) und die zweite lineare Automatenschaltung (L2) weisen eine gemeinsame Eingangsleitung zur Aufnahme eines Datenstroms  $T_n$  aus  $n$  aufeinanderfolgenden, jeweils  $k$  Bit breiten Datenwörtern  $y(1), \dots, y(n)$  auf,
- die erste lineare Automatenschaltung (L1) ist durch  
10  
15 folgende Gleichung beschreibbar,

$$z(t+1) = Az(t) \oplus y(t)$$

- die zweite lineare Automatenschaltung (L2) ist durch  
20 folgende Gleichung beschreibbar,

$$z(t+1) = Bz(t) \oplus y(t)$$

wobei  $A$  und  $B$  die Zustandsmatrizen der linearen Automatenschaltungen (L1, L2) darstellen, wobei die Zustandsmatrizen  $A$  und  $B$  invertierbar sind und wobei die Dimension  $L$  der Zustandsvektoren  $\geq k$  ist,  
25

- die erste lineare Automatenschaltung (L1) und die zweite lineare Automatenschaltung (L2) sind so ausgebildet,  
30 dass eine erste Signatur (S1) bzw. eine zweite Signatur (S2) berechenbar ist,
- $L$  nach der ersten linearen Automatenschaltung (L1) angeordnete erste Verknüpfungsgatter ( $XOR_{L1}$ ) sowie  $L$  nach

der zweiten linearen Automatenschaltung (L2) angeordnete zweite Verknüpfungsgatter ( $XOR_{L2}$ ),

- die Verknüpfungsgatter ( $XOR_{L1}$ ,  $XOR_{L2}$ ) sind so ausgebildet, dass die jeweils von der linearen Automatenschaltung (L1, L2) berechnete Signatur (S1, S2) mit einer vorgebbaren Gutsignatur vergleichbar und ein Vergleichswert ausgebbar ist.

2. Auswerteschaltung nach Anspruch 1,

dadurch gekennzeichnet, dass die Verknüpfungsgatter ( $XOR_{L1}$ ,  $XOR_{L2}$ ) als exklusive Oder-Gatter vorliegen, deren erste Eingänge jeweils mit den Ausgängen der zugehörigen linearen Automatenschaltung (L1, L2) verbunden sind und an deren zweiten Eingängen Gutsignaturen anlegbar sind.

3. Auswerteschaltung nach Anspruch 1 oder 2,

dadurch gekennzeichnet, dass vor der ersten linearen Automatenschaltung (L1) ein erster Kodierer (C1) angeordnet ist, der das Datenwort  $y(i)$  mit der Datenwortlänge von  $k$  Bit in ein kodiertes Datenwort  $u^1(i)$ ,  $u^1(i) = \text{Cod1}(y(i))$  der Wortbreite von  $K1$  Bit kodiert für  $i=1, \dots, n$ , und wobei Cod1 die Kodierungsfunktion des ersten Kodierers (C1) darstellt.

4. Auswerteschaltung nach Anspruch 3,

dadurch gekennzeichnet, dass für die Kodierungsfunktion des ersten Kodierers (C1) folgendes gilt:

$$\text{Cod1}(y'(i)) = u^1(i) \oplus f_1(e(i)),$$

oder



$$\text{Cod1}(y'(i)) = \text{Cod1}(y(i) \oplus e(i)) = \text{Cod1}(y(i) \oplus f_1(e(i)))$$

wobei für  $y'(i) = y(i) \oplus e(i)$  eine Funktion  $f_1$  mit  
 5  $f_1(0) = 0$  existiert und wobei eine Funktion  $f_1^{-1}$  mit

$$f_1^{-1}(f_1(e)) = e$$

für alle binären Datenworte  $e$  der Wortbreite  $k$ , die als  
 10 Fehler eines Datenworts vorkommen können, existiert, wobei  
 $e$  ein fehlerhaftes Datenwort des Datenstroms  $T_n$  bezeichnet.

5. Auswerteschaltung nach einem der Ansprüche 1 bis 4,  
 dadurch gekennzeichnet, dass  
 15 vor der zweiten linearen Automatenschaltung (L2) ein zweiter  
 Kodierer (C2) angeordnet ist, der das Datenwort  $y(i)$   
 mit der Datenwortlänge von  $k$  Bit in ein kodiertes Daten-  
 wort  $u^2(i)$ ,  $u^2(i) = \text{Cod2}(y(i))$  der Wortbreite von  $K_2$  Bit ko-  
 diert für  $i=1, \dots, n$ , und wobei  $\text{Cod2}$  die Kodierungsfunk-  
 20 tion des zweiten Kodierers (C2) darstellt.

6. Auswerteschaltung nach Anspruch 5,  
 dadurch gekennzeichnet, dass  
 für die Kodierungsfunktion des zweiten Kodierers (C2) fol-  
 25 gendes gilt:

$$\text{Cod2}(y'(i)) = u^2(i) \oplus f_2(e(i))$$

oder

30

$$\begin{aligned} \text{Cod2}(y'(i)) &= \text{Cod2}(y(i) \oplus e(i)) \\ &= \text{Cod2}(y(i)) \oplus f_2(e(i)) \end{aligned}$$

wobei eine Funktion  $f_2^{-1}$  mit

$$f_2^{-1}(f_2(e)) = e$$

5

für alle binären Datenworte  $e$  der Wortbreite  $k$ , die als Fehler eines Datenworts vorkommen können, existiert, wobei  $e$  ein fehlerhaftes Datenwort des Datenstroms  $T_n$  bezeichnet.

- 10 7. Auswerteschaltung nach einem der Ansprüche 3 bis 6, dadurch gekennzeichnet, dass die Wortbreite  $K1$  der von dem ersten Kodierer (C1) kodierten Datenwörter  $u^1(i)$  gleich der Wortbreite  $K2$  der von dem zweiten Kodierer (C2) kodierten Datenwörter  $u^2(i)$  ist.

15

8. Auswerteschaltung nach einem der Ansprüche 3 bis 7, dadurch gekennzeichnet, dass der erste Kodierer (C1) hinsichtlich seines Aufbaus und seiner Funktion mit dem zweiten Kodierer (C2) übereinstimmt.

20

9. Auswerteschaltung nach einem der Ansprüche 3 bis 8, dadurch gekennzeichnet, dass die Wortbreite  $K1$  der von dem ersten Kodierer (C1) kodierten Datenwörter  $u^1(i)$  und die Wortbreite  $K2$  der von dem zweiten Kodierer (C2) kodierten Datenwörter  $u^2(i)$  jeweils gleich der Wortbreite  $k$  der Datenwörter  $y(1), \dots, y(n)$  des Datenstroms  $T_n$  ist.

25

- : 30 10. Auswerteschaltung nach einem der Ansprüche 3 bis 9, dadurch gekennzeichnet, dass

die Kodierungsfunktionen Cod1 und Cod2 des ersten Kodierers und des zweiten Kodierers (C2) wie folgt ausgebildet sind:

$$\begin{aligned} & \text{Cod1}(y_1(i), y_2(i), \dots, y_k(i)) \\ &= P1(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0) \end{aligned}$$

$$\begin{aligned} & \text{Cod2}(y_1(i), y_2(i), \dots, y_k(i)) \\ &= P2(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0) \end{aligned}$$

10

für  $i, 1, \dots, n$

wobei die Anzahl der am Ende von  $P1(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  befindlichen Nullen gleich  $(K1-k)$  ist, wobei die Anzahl der am Ende von  $P2(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  gleich  $(K2-k)$  ist und wobei P1 eine beliebige Permutation der K1 Komponenten von  $(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  und P2 eine beliebige Permutation der K2 Komponenten von  $(y_1(i), y_2(i), \dots, y_k(i), 0, \dots, 0)$  darstellen.

20

11. Auswerteschaltung nach einem der Ansprüche 3 bis 9, dadurch gekennzeichnet, dass die Kodierungsfunktionen Cod1 und Cod2 des ersten Kodierers und des zweiten Kodierers (C2) wie folgt ausgebildet sind:

25

$$\begin{aligned} & \text{Cod1}(y_1(i), y_2(i), \dots, y_k(i)) \\ &= P1(y_1(i), y_2(i), \dots, y_k(i), b_1^1, \dots, b_{K1-k}^1) \end{aligned}$$

30

$$\begin{aligned} & \text{Cod2}(y_1(i), y_2(i), \dots, y_k(i)) \\ &= P2(y_1(i), y_2(i), \dots, y_k(i), b_1^2, \dots, b_{K2-k}^2) \end{aligned}$$

mit  $b_1^1, \dots, b_{K_1-k}^1, b_1^2, \dots, b_{K_2-k}^2 \in \{0,1\}$  und wobei P1 und P2 beliebige Permutationen darstellen.

- 5 12. Auswerteschaltung nach einem der Ansprüche 3 bis 11,  
dadurch gekennzeichnet, dass  
die Kodierungsfunktion Cod1 des ersten Kodierers (C1) so  
ausgebildet ist, dass sie einen linearen Blockkode,  
f<sub>1</sub>=Cod1, realisiert.
- 10 13. Auswerteschaltung nach einem der Ansprüche 3 bis 9 oder  
nach Anspruch 12,  
dadurch gekennzeichnet, dass  
die Kodierungsfunktion Cod2 des zweiten Kodierers (C2) so  
15 ausgebildet ist, dass sie einen linearen Blockkode,  
f<sub>2</sub>=Cod2, realisiert.
14. Auswerteschaltung nach einem der Ansprüche 1 bis 13,  
dadurch gekennzeichnet, dass  
20 die Zustandsmatrix A der ersten linearen Automaten-  
schaltung (L1) und die Zustandsmatrix B der zweiten linearen  
Automatenschaltung (L2) wie folgt miteinander in Beziehung  
stehen:
- 25 
$$B = A^n$$
- mit  $n \neq 1$ .
15. Auswerteschaltung nach einem der Ansprüche 1 bis 14,  
30 dadurch gekennzeichnet, dass

die Zustandsmatrix B der zweiten linearen Automatenschaltung (L2) gleich der invertierten Zustandsmatrix  $A^{-1}$  der ersten linearen Automatenschaltung (L1) ist

- 5 16. Auswerteschaltung nach einem der Ansprüche 1 bis 15,  
dadurch gekennzeichnet, dass  
die erste lineare Automatenschaltung (L1) als linear rück-  
gekoppeltes Schieberegister und die zweite lineare Automa-  
tenschaltung (L2) als inverses linear rückgekoppeltes  
10 Schieberegister ausgebildet sind, wobei beide lineare Au-  
tomatenschaltungen (L1, L2) eine parallele Eingabe aufwei-  
sen.
17. Auswerteschaltung nach einem der Ansprüche 1 bis 16,  
15 dadurch gekennzeichnet, dass  
die erste lineare Automatenschaltung (L1) als linear rück-  
gekoppeltes, K1-dimensionales Multi-Input-Schieberegister  
und/oder die zweite lineare Automatenschaltung (L2) als  
linear rückgekoppeltes, K2-dimensionales Multi-Input-  
20 Schieberegister ausgebildet sind.
18. Auswerteschaltung nach Anspruch 17,  
dadurch gekennzeichnet, dass  
das/die Multi-Input-Schieberegister (L1, L2) ein primiti-  
25 ves Rückkopplungspolynom maximaler Länge aufweist/auf-  
weisen.
19. Verfahren zum Feststellen und/oder zum Lokalisieren von  
fehlerhaften Datenworten in einem Datenstrom  $T_n$ , wobei das  
30 Verfahren die folgenden Verfahrensschritte aufweist:  
a) Eingeben von Datenwörtern  $y(1), \dots, y(i-1), y'(i),$   
 $y(i+1), \dots, y(n)$  eines Datenstroms  $T_n$  in einen ersten  
Kodierer (C1),

b) Kodieren der Datenwörter  $y(1), \dots, y(n)$  in kodierte Datenwörter  $u^1(1), \dots, u^1(n)$  der Wortbreite  $K_1$  mit  $K_1 \geq k$  mittels der Kodierfunktion  $Cod_1$  des ersten Kodierers ( $C_1$ ),

5 c) Eingeben der kodierten Datenwörter  $u^1(1), \dots, u^1(i-1), u^{1'}(i)$  oder  $u^1(i), u^1(i), \dots, u^1(n)$  in die Eingänge einer ersten linearen Automatenschaltung ( $L_1$ ), die durch die Automaten Gleichung

10 
$$z^1(t+1) = A \cdot z^1(t) + u^1(t)$$

beschrieben ist, wobei  $z^1$  einen  $K_1$ -dimensionaler Zustandsvektor und  $A$  eine  $K_1 \times K_1$ -Zustandsmatrix darstellen, und wobei die Zustandsmatrix  $A$  invertierbar ist,

15 d) Verarbeiten der kodierten Datenwörter  $u^1(1), \dots, u^1(i-1), u^{1'}(i)$  oder  $u^1(i), u^1(i), \dots, u^1(n)$  durch die erste lineare Automatenschaltung ( $L_1$ ), wobei die erste lineare Automatenschaltung ( $L_1$ )

- in den Zustand  $z^1(n+1) = S_1(L_1, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  übergeht, wenn bei den kodierten Datenwörtern  $u^1(1), \dots, u^1(i-1), u^1(i), u^1(i+1), \dots, u^1(n)$  kein Fehler feststellbar ist,

20 -  $z^{1'}(n+1) = S_1(L_1, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$  übergeht, wenn wenigstens bei der  $i$ -ten Position der kodierten Datenwörter  $u^1(1), \dots, u^1(i-1), u^{1'}(i), \dots, u^1(n)$  ein Fehler vorliegt,

25 wobei die Signatur eines fehlerfreien Datenstroms  $T_n$  mit  $S(L_1, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  und die Signatur eines fehlerhaften Datenstroms  $T_n$  mit  $S(L_1, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$  bezeichnet sind.

30 e) Überprüfen der bestimmten Signatur des Datenstroms  $T_n$  und Fortfahren mit Verfahrensschritt a) für weitere Da-

tenströme  $T_n$ , falls es sich bei der bestimmten Signatur des Datenstroms  $T_n$  um die Signatur eines fehlerfreien Datenstroms  $T_n$  handelt,

- f) Eingeben der Datenwörter  $y(1), \dots, y(i-1), y'(i), \dots, y(n)$  des Datenstromes  $T_n$  in einem zweiten Kodierer (C2),
- g) Kodieren der Datenwörter  $y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n)$  in kodierte Datenwörter  $u^2(1), \dots, u^2(i-1), u^{2'}(i)$  oder  $u^2(i), u^2(i), \dots, u^2(n)$  der Wortbreite  $K_2$  mit  $K_2 \geq k$  mittels der Kodierfunktion  $Cod_2$  des zweiten Kodierers (C2),
- h) Eingeben der kodierten Datenwörter  $u^2(1), \dots, u^2(i-1), u^{2'}(i)$  oder  $u^2(i), u^2(i), \dots, u^2(n)$  in die Eingänge einer zweiten linearen Automaten-schaltung (L2), die durch die Automaten-gleichung

$$z^2(t+1) = B \cdot z^2(t) \oplus u^2(t)$$

beschrieben ist, wobei  $z^2$  einen  $K_2$ -dimensionalen Zustandsvektor und  $B$  eine  $K_2 \times K_2$ -Zustandsmatrix mit  $B \neq A$  darstellen, und wobei die Zustandsmatrix  $B$  invertierbar ist,

- i) Verarbeiten der kodierten Datenwörter  $u^2(1), \dots, u^2(i-1), u^{2'}(i)$  oder  $u^2(i), u^2(i), \dots, u^2(n)$  durch die zweite lineare Automaten-schaltung (L2), wobei die zweite lineare Automaten-schaltung (L2)
- in den Zustand  $z^2(n+1) = S_2(L_2, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  übergeht, wenn bei den Datenwörtern  $u^2(1), \dots, u^2(i-1), u^2(i), u^2(i), \dots, u^2(n)$  kein Fehler feststellbar ist,
  - $z^{2'}(n+1) = S_2(L_2, y(1), \dots, y(i-1), y(i), y'(i), y(i+1), \dots, y(n))$  übergeht, wenn wenigstens bei der  $i$ -ten Position der kodierten Datenwörter  $u^2(1), \dots,$



$u^2(i-1)$ ,  $u^{2'}(i)$ ,  $u^2(i)$ , ...,  $u^2(n)$  ein Fehler vor-  
liegt,

wobei die Signatur eines fehlerfreien Datenstroms  $T_n$  mit  
 $S(L2, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n))$  und  
5 die Signatur eines fehlerhaften Datenstroms  $T_n$  mit  
 $S(L2, y(1), \dots, y(i-1), y'(i), \dots, y(n))$  bezeichnet  
sind.

- j) Bestimmen der Signaturdifferenzen  $\Delta S1$  und  $\Delta S2$  durch ex-  
klusive Oder-Verknüpfungen der in Verfahrensschritt d)  
10 bzw. i) bestimmten Signaturen  $S1$  und  $S2$  mit ermittelten  
Gutsignaturen, jeweils nach folgenden Vorschriften:

$$\Delta S1 = S(L1, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)) \\ \oplus S(L1, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$$

$$\Delta S2 = S(L2, y(1), \dots, y(i-1), y(i), y(i+1), \dots, y(n)) \\ \oplus S(L2, y(1), \dots, y(i-1), y'(i), y(i+1), \dots, y(n))$$

- k) Bestimmen einer eindeutigen Lösung für die Position  $i$   
20 des fehlerhaften Bits im fehlerhaften Datenwort durch  
Lösen der Gleichung

$$f_1^{-1}(A^{i-n} \Delta S1) = f_2^{-1}(B^{i-n} \Delta S2)$$

25 falls sich keine eindeutige Lösung für  $1 \leq i \leq n$  er-  
gibt, Ausgeben einer Mitteilung durch ein Ausgabemedi-  
um, dass in dem betrachteten Datenstroms  $T_n$  zwei oder  
mehr Fehler vorliegen,

- l) Bestimmen einer eindeutigen Lösung für den Zähler  $e(i)$   
30 des fehlerhaften Datenworts  $y'(i)$  im Datenstrom  $T_n$  durch  
Lösen der Gleichung

$$e(i) = f_1^{-1}(A^{i-n} \cdot \Delta S1)$$

- 5 m) Ausgeben der Position  $i$  des fehlerhaften Bits im fehlerhaften Datenwort sowie des Fehlers  $e(i)$  des fehlerhaften Datenworts  $y'(i)$  im Datenstroms  $T_n$  durch ein Ausgabemedium.
20. Verfahren nach Anspruch 19,  
dadurch gekennzeichnet, dass  
10 die Verfahrensschritte mit einer Auswerteschaltung gemäß einem der Ansprüche 1 bis 18 durchgeführt werden.
21. Auswerteschaltung nach einem der Ansprüche 1 bis 18,  
dadurch gekennzeichnet, dass  
15 die Auswerteschaltung auf einem integrierten Schaltkreis monolithisch integriert ist.
22. Loadboard zur Aufnahme wenigstens einer Nadelkarte zum Testen von integrierten Schaltkreisen und/oder mit wenigstens einem Testsockel zum Testen von integrierten Schaltkreisen und/oder zum Anschluss eines Handlers an einen Tester von integrierten Schaltkreisen, wobei das Loadboard eine Auswerteschaltung nach einem der Ansprüche 1 bis 18 aufweist.  
25
23. Nadelkarte zum Testen von integrierten Schaltungen, bei der eine Auswerteschaltung nach einem der Ansprüche 1 bis 18 integriert ist.
- 30 24. Tester zum Testen von integrierten Schaltkreisen mit den folgenden Merkmalen:

- der Tester verfügt über mehrere Instrumente zum Erzeugen von Signalen oder Datenströmen und über mehrere Mess-Sensoren, insbesondere für Ströme und Spannungen,
- der Tester weist ein Loadboard auf, das zur Aufnahme wenigstens einer Nadelkarte zum Testen von integrierten Schaltkreisen und/oder zum Anschluss eines Handlers an einen Tester von integrierten Schaltkreisen vorgesehen ist und/oder das mit wenigstens einem Testsockel zum Testen von integrierten Schaltkreisen ausgestattet ist,
- der Tester weist eine Auswerteschaltung nach einem der Ansprüche 1 bis 18 auf.

25. Computerprogramm zum Ausführen eines Verfahrens zum Feststellen und/oder zum Lokalisieren von fehlerhaften Datenworten in einem Datenstrom  $T_n$ , das so ausgebildet ist, daß wenigstens die Verfahrensschritte a) sowie k), l) und m) gemäß Anspruch 19 oder 20 ausführbar sind.

26. Computerprogramm nach Anspruch 25, das auf einem Speichermedium, insbesondere in einem Computerspeicher oder in einem Direktzugriffsspeicher enthalten ist.

27. Computerprogramm nach Anspruch 25, das auf einem elektrischen Trägersignal übertragen wird.

28. Datenträger mit einem Computerprogramm nach Anspruch 25.

29. Verfahren, bei dem ein Computerprogramm nach Anspruch 24 aus einem elektronischen Datennetz wie bspw. aus dem Internet auf einen an das Datennetz angeschlossenen Computer heruntergeladen wird.

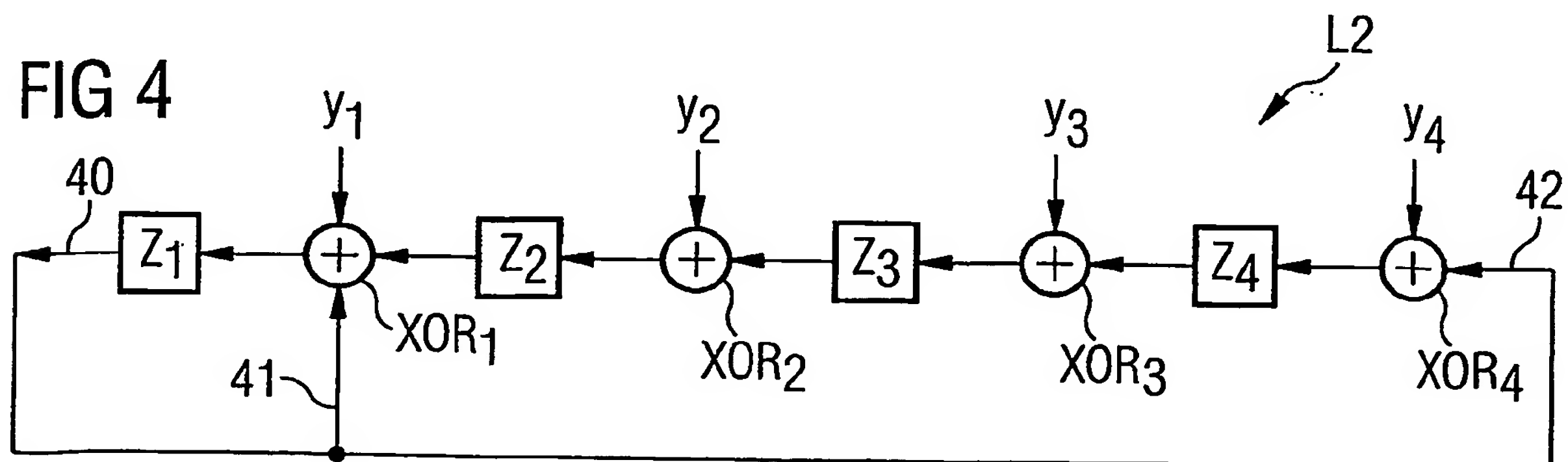
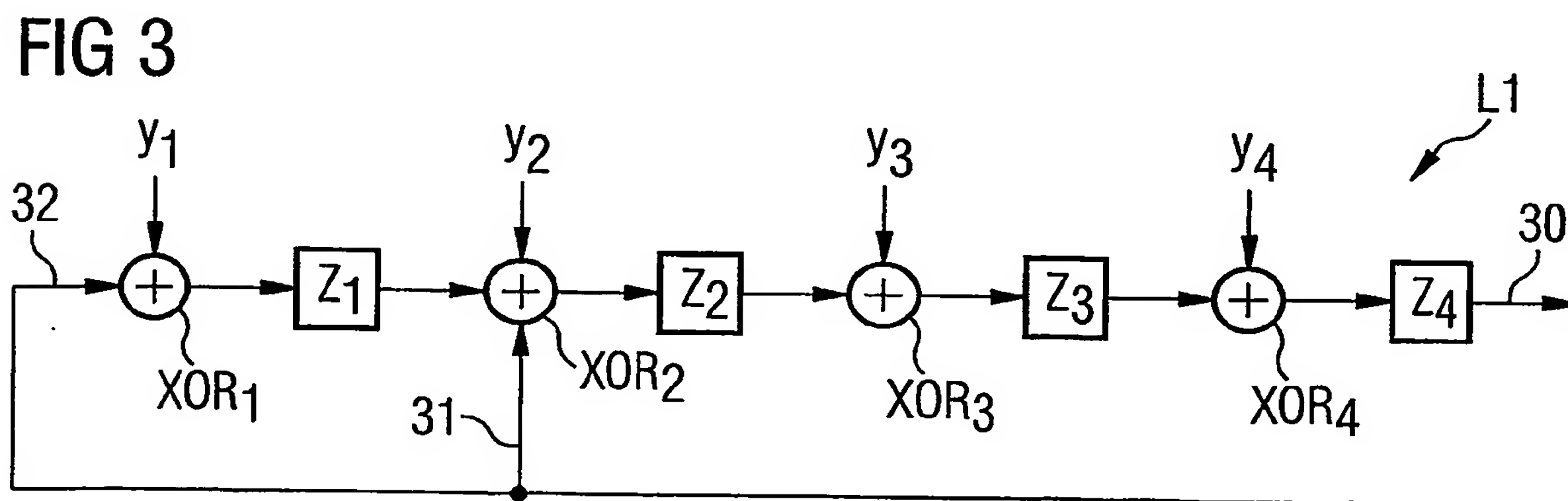
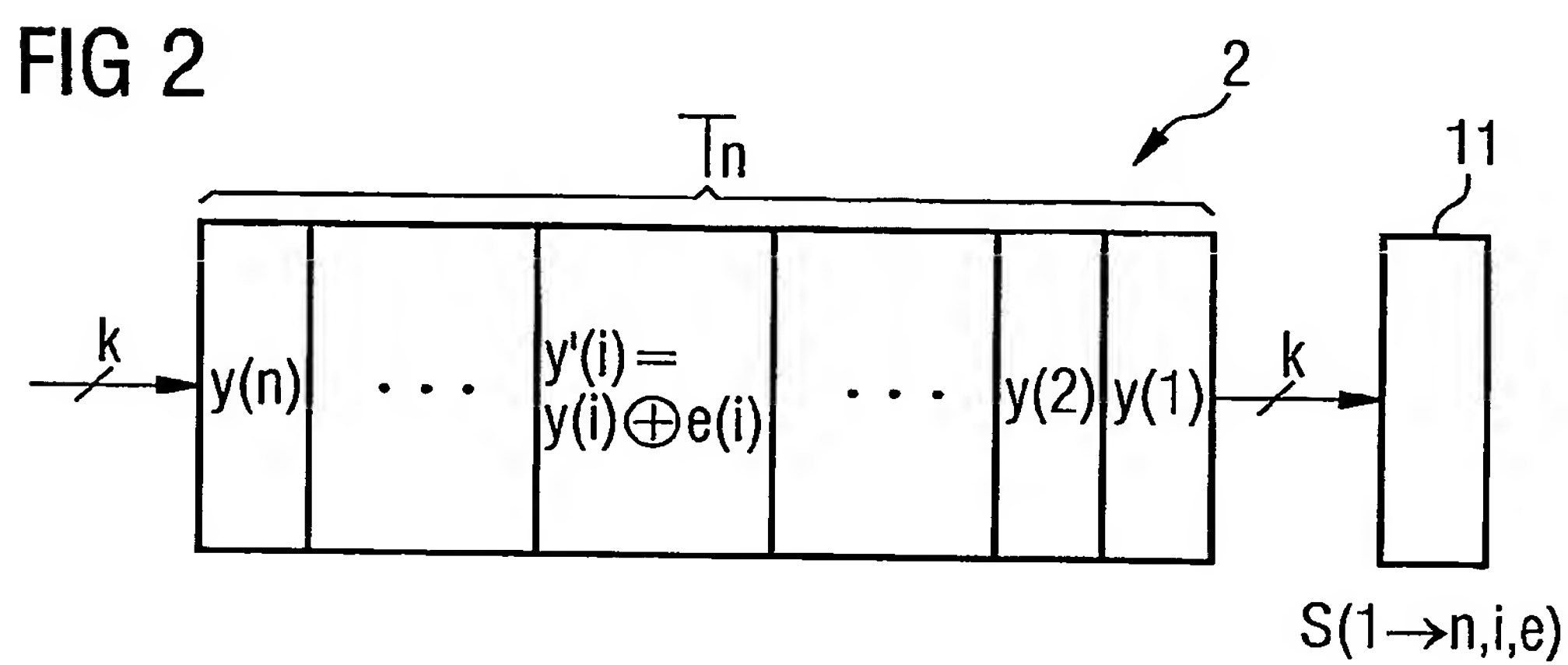
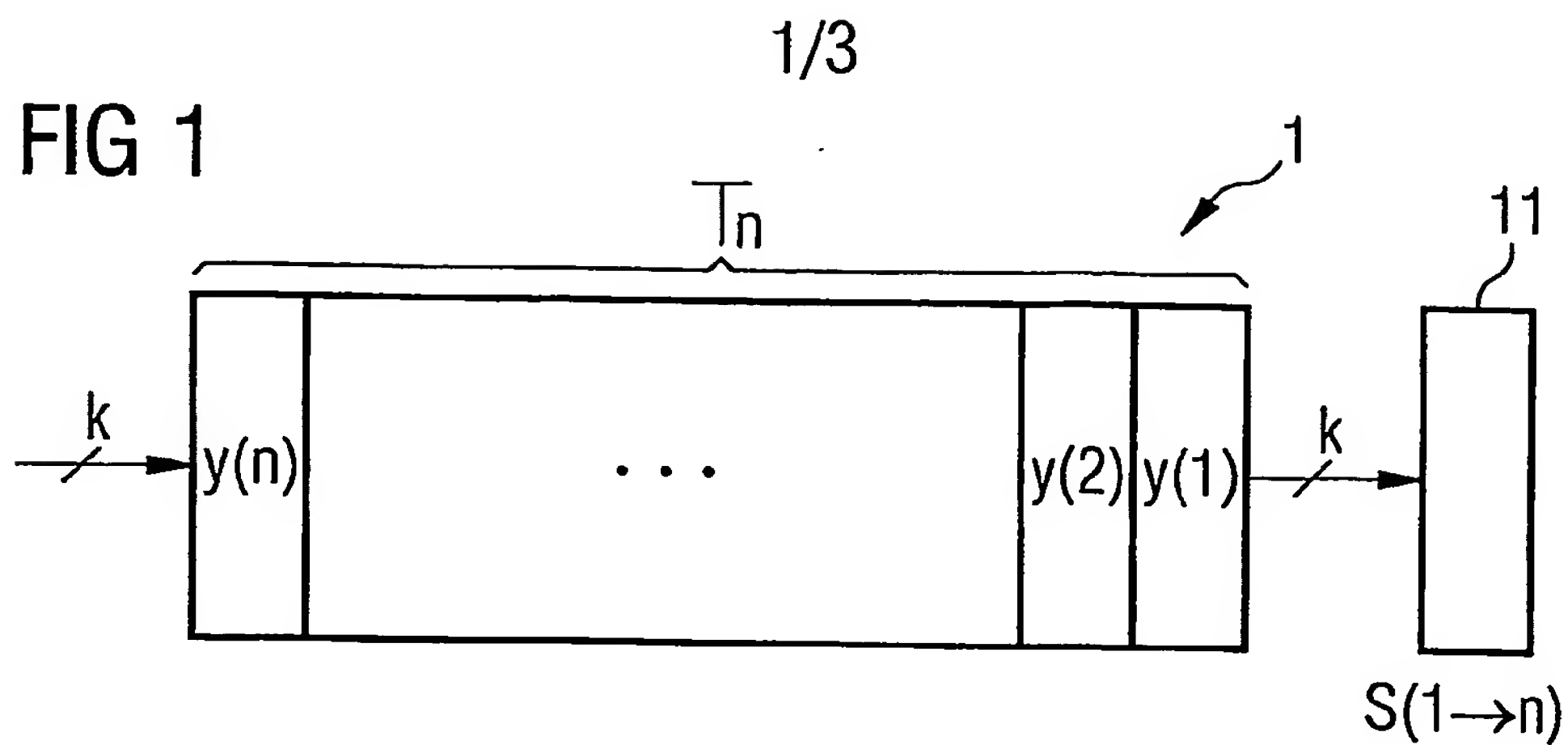


FIG 5

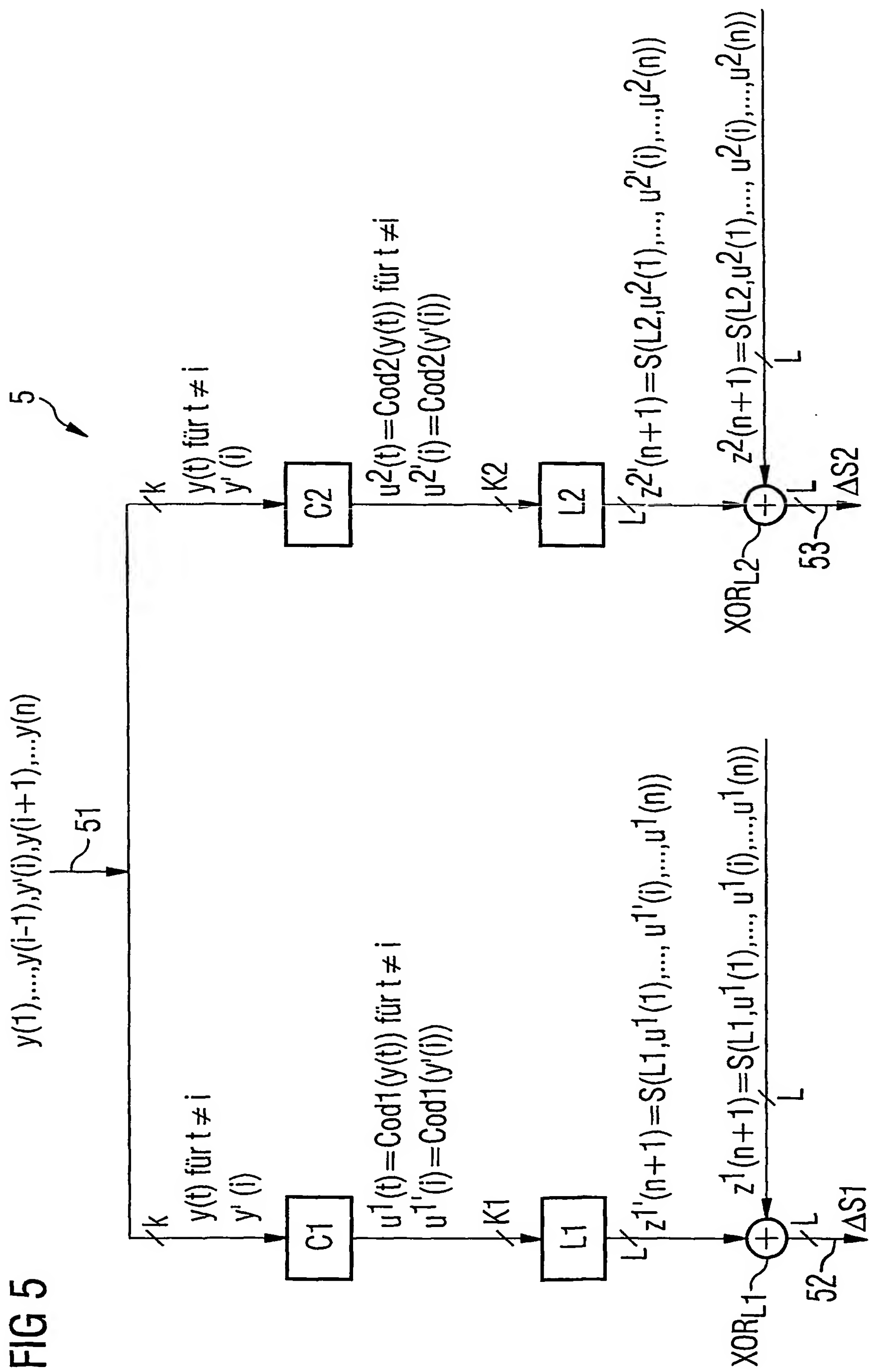
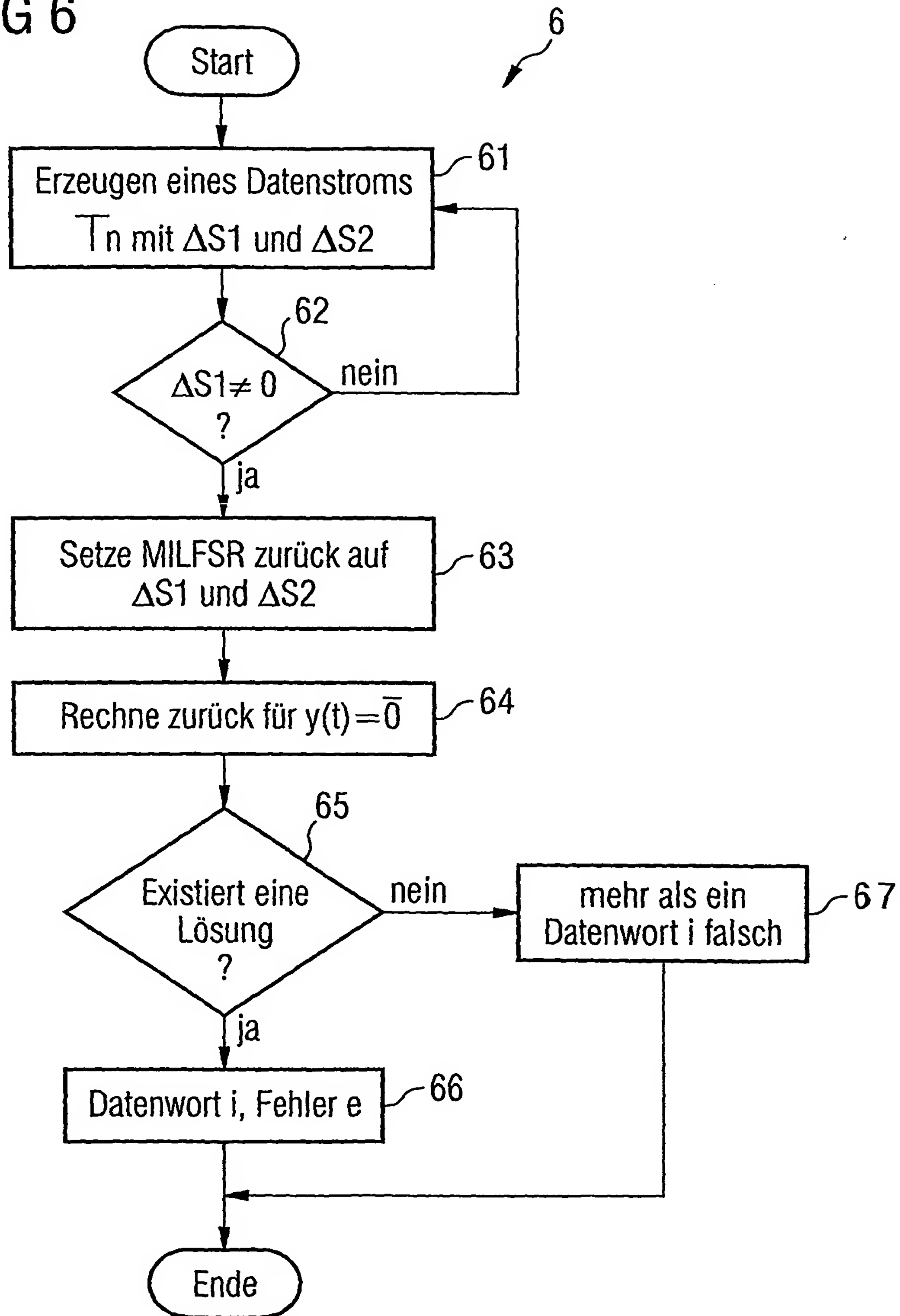


FIG 6



# INTERNATIONAL SEARCH REPORT

International Application No  
PCT/DE 2004/002362

## A. CLASSIFICATION OF SUBJECT MATTER

IPC 7 G01R31/3193

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC 7 G01R

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the International search (name of data base and, where practical, search terms used)

EPO-Internal, INSPEC

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>SIGNATURE ANALYSIS FOR IDENTIFYING FAILING VECTORS, ITG-GMM, 15. WORKSHOP, TESTMETHODS AND RELIABILITY OF CIRCUITS AND SYSTEMS, 'Online! 25 March 2003 (2003-03-25), XP002317738 BREMEN, GERMANY ITG-GMM, 15. Workshop, Testmethods and Reliability of Circuits and Systems Retrieved from the Internet: URL: <a href="http://www.item.uni-bremen.de/itg-fg5/workshop_15/paper/goessel_analyzingFailingVectors.pdf">http://www.item.uni-bremen.de/itg-fg5/workshop_15/paper/goessel_analyzingFailingVectors.pdf</a> 'retrieved on 2005-02-15! the whole document</p> <p style="text-align: center;">----- -/--</p>	1-29



Further documents are listed in the continuation of box C.



Patent family members are listed in annex.

### \* Special categories of cited documents:

- \*A\* document defining the general state of the art which is not considered to be of particular relevance
- \*E\* earlier document but published on or after the International filing date
- \*L\* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- \*O\* document referring to an oral disclosure, use, exhibition or other means
- \*P\* document published prior to the international filing date but later than the priority date claimed

- \*T\* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- \*X\* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- \*Y\* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.
- \*G\* document member of the same patent family

Date of the actual completion of the international search

15 February 2005

Date of mailing of the international search report

03/03/2005

Name and mailing address of the ISA

European Patent Office, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax (+31-70) 340-3016

Authorized officer

Meggyesi, Z



# INTERNATIONAL SEARCH REPORT

International Application No  
PCT/DE2004/002362

**C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT**

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	<p>GOESSEL M ET AL: "A PARITY-PRESERVING MULTI-INPUT SIGNATURE ANALYZER AND ITS APPLICATION FOR CONCURRENT CHECKING AND BIST" JOURNAL OF ELECTRONIC TESTING, KLUWER ACADEMIC PUBLISHERS, DORDRECHT, NL, vol. 8, no. 2, 1 April 1996 (1996-04-01), pages 165-176, XP000598709 ISSN: 0923-8174 the whole document</p>	1-29
A	<p>US 5 831 992 A (WU ET AL) 3 November 1998 (1998-11-03) column 3, line 35 - column 4, line 28; claim 1; figure 1</p>	1-29

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No

PCT/DE2004/002362

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5831992	A	03-11-1998	NONE

# INTERNATIONALER RECHERCHENBERICHT

Internationales Aktenzeichen

PCT/DE2004/002362

## A. KLASSIFIZIERUNG DES ANMELDUNGSGEGENSTANDES

IPK 7 601R31/3193

Nach der Internationalen Patentklassifikation (IPK) oder nach der nationalen Klassifikation und der IPK

## B. RECHERCHIERTE GEBIETE

Recherchierte Mindestprüfstoff (Klassifikationssystem und Klassifikationssymbole)

IPK 7 601R

Recherchierte aber nicht zum Mindestprüfstoff gehörende Veröffentlichungen, soweit diese unter die recherchierten Gebiete fallen

Während der internationalen Recherche konsultierte elektronische Datenbank (Name der Datenbank und evtl. verwendete Suchbegriffe)

EPO-Internal, INSPEC

## C. ALS WESENTLICH ANGESEHENE UNTERLAGEN

Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Beitr. Anspruch Nr.
A	<p>SIGNATURE ANALYSIS FOR IDENTIFYING FAILING VECTORS, ITG-GMM, 15. WORKSHOP, TESTMETHODS AND RELIABILITY OF CIRCUITS AND SYSTEMS, 'Online! 25. März 2003 (2003-03-25), XP002317738 BREMEN, GERMANY ITG-GMM, 15. Workshop, Testmethods and Reliability of Circuits and Systems Gefunden im Internet: URL: <a href="http://www.item.uni-bremen.de/itg-fg5/workshop_15/paper/goessel_analyzingFailingVectors.pdf">http://www.item.uni-bremen.de/itg-fg5/workshop_15/paper/goessel_analyzingFailingVectors.pdf</a> 'gefunden am 2005-02-15! das ganze Dokument</p> <p style="text-align: center;">----- -/--</p>	1-29

☒ Weitere Veröffentlichungen sind der Fortsetzung von Feld C zu entnehmen

☒ Siehe Anhang Patentfamilie

- \* Besondere Kategorien von angegebenen Veröffentlichungen :
- \*A\* Veröffentlichung, die den allgemeinen Stand der Technik definiert, aber nicht als besonders bedeutsam anzusehen ist
  - \*E\* älteres Dokument, das jedoch erst am oder nach dem internationalen Anmeldedatum veröffentlicht worden ist
  - \*L\* Veröffentlichung, die geeignet ist, einen Prioritätsanspruch zweifelhaft erscheinen zu lassen, oder durch die das Veröffentlichungsdatum einer anderen im Recherchenbericht genannten Veröffentlichung belegt werden soll oder die aus einem anderen besonderen Grund angegeben ist (wie ausgeführt)
  - \*O\* Veröffentlichung, die sich auf eine mündliche Offenbarung, eine Benutzung, eine Ausstellung oder andere Maßnahmen bezieht
  - \*P\* Veröffentlichung, die vor dem internationalen Anmeldedatum, aber nach dem beanspruchten Prioritätsdatum veröffentlicht worden ist

- \*T\* Spätere Veröffentlichung, die nach dem internationalen Anmeldedatum oder dem Prioritätsdatum veröffentlicht worden ist und mit der Anmeldung nicht kollidiert, sondern nur zum Verständnis des der Erfindung zugrundeliegenden Prinzips oder der ihr zugrundeliegenden Theorie angegeben ist
- \*X\* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann allein aufgrund dieser Veröffentlichung nicht als neu oder auf erfinderischer Tätigkeit beruhend betrachtet werden
- \*Y\* Veröffentlichung von besonderer Bedeutung; die beanspruchte Erfindung kann nicht als auf erfinderischer Tätigkeit beruhend betrachtet werden, wenn die Veröffentlichung mit einer oder mehreren anderen Veröffentlichungen dieser Kategorie in Verbindung gebracht wird und diese Verbindung für einen Fachmann naheliegend ist
- \*Z\* Veröffentlichung, die Mitglied derselben Patentfamilie ist

Datum des Abschlusses der internationalen Recherche

15. Februar 2005

Absenddatum des internationalen Recherchenberichts

03/03/2005

Name und Postanschrift der Internationalen Recherchenbehörde  
Europäisches Patentamt, P.B. 5818 Patentlaan 2  
NL - 2280 HV Rijswijk  
Tel. (+31-70) 340-2040, Tx. 31 651 epo nl,  
Fax (+31-70) 340-3016

Bevollmächtigter Bediensteter

Meggyesi, Z

INTERNATIONALER RECHERCHENBERICHT

Internationales Aktenzeichen  
PCT/DE2004/002362

C.(Fortsetzung) ALS WESENTLICH ANGESEHENE UNTERLAGEN		
Kategorie*	Bezeichnung der Veröffentlichung, soweit erforderlich unter Angabe der in Betracht kommenden Teile	Betr. Anspruch Nr.
A	GOESSEL M ET AL: "A PARITY-PRESERVING MULTI-INPUT SIGNATURE ANALYZER AND ITS APPLICATION FOR CONCURRENT CHECKING AND BIST" JOURNAL OF ELECTRONIC TESTING, KLUWER ACADEMIC PUBLISHERS, DORDRECHT, NL, Bd. 8, Nr. 2, 1. April 1996 (1996-04-01), Seiten 165-176, XP000598709 ISSN: 0923-8174 das ganze Dokument	1-29
A	US 5 831 992 A (WU ET AL) 3. November 1998 (1998-11-03) Spalte 3, Zeile 35 - Spalte 4, Zeile 28; Anspruch 1; Abbildung 1	1-29

Angaben zu Veröffentlichungen, die zur selben Patentfamilie gehören

PCT/DE2004/002362

Formblatt PCT/ISA/210 (Anhang Patentfamilie) (Januar 2004)